

Machine learning discovery of new phases in programmable quantum simulator snapshots

Cole Miles¹, Rhine Samajdar², Sepehr Ebadi², Tout T. Wang², Hannes Pichler^{3,4}, Subir Sachdev^{2,5}, Mikhail D. Lukin², Markus Greiner², Kilian Q. Weinberger⁶, and Eun-Ah Kim^{1,*}

¹Department of Physics, Cornell University, Ithaca, New York 14853, USA

²Department of Physics, Harvard University, Cambridge, Massachusetts 02138, USA

³Institute for Theoretical Physics, University of Innsbruck, Innsbruck A-6020, Austria

⁴Institute for Quantum Optics and Quantum Information, Austrian Academy of Sciences, Innsbruck A-6020, Austria

⁵School of Natural Sciences, Institute for Advanced Study, Princeton, New Jersey 08540, USA

⁶Department of Computer Science, Cornell University, Ithaca, New York 14853, USA



(Received 23 December 2021; accepted 22 December 2022; published 19 January 2023)

Machine learning has recently emerged as a promising approach for studying complex phenomena characterized by rich datasets. In particular, data-centric approaches lead to the possibility of automatically discovering structures in experimental datasets that manual inspection may miss. Here, we introduce an interpretable unsupervised-supervised hybrid machine learning approach, the hybrid-correlation convolutional neural network (hybrid-CCNN), and apply it to experimental data generated using a programmable quantum simulator based on Rydberg atom arrays. Specifically, we apply hybrid-CCNN to discover and identify new quantum phases on square lattices with programmable interactions. The initial unsupervised dimensionality reduction and clustering stage first reveals five distinct quantum phase regions. In a second supervised stage, we refine these phase boundaries and seek insights into the phases by training multiple CCNN classifiers. A learned spatial weighting, introduced to the CCNNs in this work, enables discovery of spatial structure at scales beyond the filter size. The characteristic spatial weightings and snippets of correlations specifically recognized in each phase capture quantum fluctuations in the striated phase and identify a previously undetected boundary-ordered phase as well as motifs of more exotic ordered phases. These observations demonstrate that a combination of programmable quantum simulators with machine learning can be used as a powerful tool for detailed exploration of correlated quantum states of matter.

DOI: [10.1103/PhysRevResearch.5.013026](https://doi.org/10.1103/PhysRevResearch.5.013026)

I. INTRODUCTION

Recent advances in realization of programmable quantum simulators (PQSs) have opened up a new era in the exploration of strongly correlated quantum matter [1–4], which calls for new approaches for analyzing large volumes of data generated by such quantum devices. Using optical techniques, it is possible to arrange a large number of qubits in arbitrary lattice geometries [5] and to control the Hamiltonian evolution of the system [2] dynamically in real time. Remarkably, these simulators can probe states within an extremely large Hilbert space. For example, in a 13×13 system, the quantum states live in a 2^{169} -dimensional space while each measurement probabilistically projects to just a single dimension. Tomographically [6–8] inferring the entire many-body wavefunction from such measurements themselves is a formidable task. While certain types of many-body states can be easily identified by evaluating simple local observables, many exotic quantum phases

that can be explored on programmable simulators cannot be characterized using conventional approaches.

In this paper, we introduce a hybrid unsupervised-supervised machine learning (ML) approach to analyze the data generated using programmable quantum simulators. Specifically, we apply this method to a PQS based on Rydberg atoms arrayed on a square lattice [5]. We show how the unsupervised stage automatically discovers new phases and reveals *a priori* unknown order parameters when the ordering is long ranged. Importantly, we uncover several features including: (i) the pattern of quantum correlations in the “striated phase”; (ii) a boundary-ordering quantum phase transition in which the edge of the system develops long-range order while the bulk remains trivial; and (iii) previously undetected ordering motifs corresponding to the more exotic “rhombic”, “staggered”, and “nematic” phases.

Our programmable Rydberg quantum simulator [see Fig. 1(a)] consists of neutral atoms trapped in defect-free arrays of optical tweezers with programmable geometries [5]. Coherent laser excitations to atomic Rydberg states realize an Ising-like spin model [2,4] described by the Hamiltonian

$$\frac{H}{\hbar} = \frac{\Omega}{2} \sum_i (|g_i\rangle\langle r_i| + |r_i\rangle\langle g_i|) - \Delta \sum_i n_i + \sum_{i<j} V_{ij} n_i n_j, \quad (1)$$

*eun-ah.kim@cornell.edu

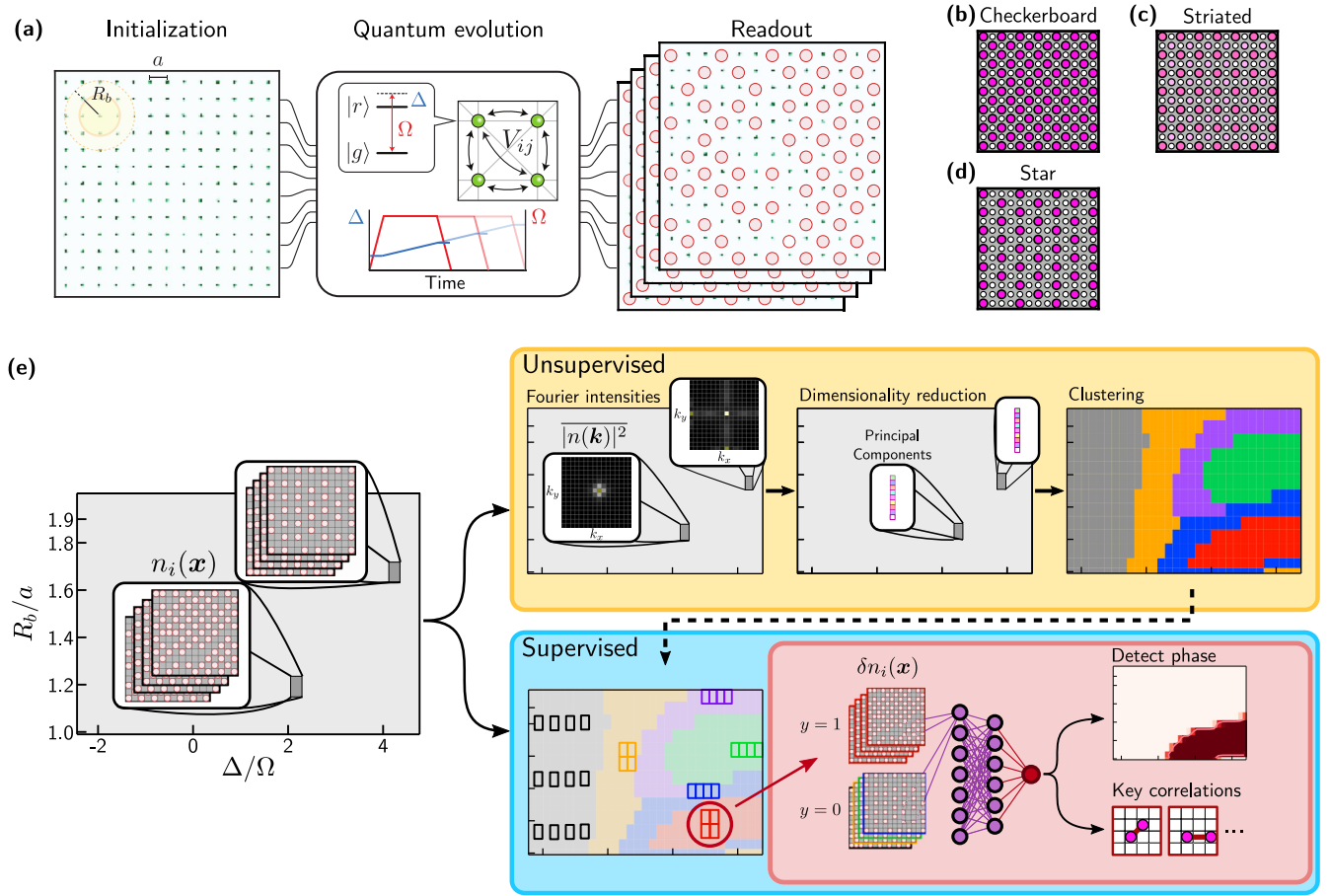


FIG. 1. (a) Defect-free square lattices of neutral atoms undergo coherent quantum evolution for different values of blockade extent R_b/a and linear detuning sweeps' endpoints Δ/Ω , followed by projective readout in which atoms excited to the Rydberg state are detected as loss (red circles). (b)–(d) Idealized real-space patterns corresponding to phases predicted to be present at various regions of parameter space. Dark pink and white sites indicate $|r\rangle$ and $|g\rangle$ states, respectively, while the light pink sites in the striated phase are in a quantum superposition of $|r\rangle$ and $|g\rangle$. (e) A diagram outlining the hybrid-CCNN approach. First, an unsupervised technique is used to generate a rough first-pass phase diagram. Here, we choose to measure average Fourier amplitudes $|n(\mathbf{k})|^2$ at each (Δ, R_b) , perform a dimensionality reduction using principal component analysis, and finally cluster using a Gaussian mixture model. The resulting phase diagram informs the starting “seeds” in the parameter space, from which snapshots are sampled in a second supervised stage. We then learn to distinguish these snapshots using interpretable classifiers, from which we can extract refined phase boundaries and key identifying features.

where atoms in the ground (Rydberg) state are denoted by $|g\rangle$ ($|r\rangle$), and $n_i \equiv |r_i\rangle\langle r_i|$. The transverse field Ω corresponds to the Rabi frequency of the laser field, the longitudinal field Δ corresponds to the laser detuning, and $V_{ij} \equiv \mathcal{V}_0/|\mathbf{x}_i - \mathbf{x}_j|^6$ is the long-range van der Waals interactions between Rydberg excitations at $\mathbf{x}_i$ and $\mathbf{x}_j$.

Density-matrix renormalization group (DMRG) calculations with a truncated interaction [9] have predicted a number of quantum phases arising from the interplay between coherent laser driving and the long-range van der Waals interactions [see Figs. 1(b)–1(e)]. These phases can be understood based on the Rydberg blockade phenomenon [10]: The strong interactions V_{ij} can prohibit (or “blockade”) the simultaneous excitation of neighboring atoms to the Rydberg state. The spatial extent of this blockade (or equivalently, the interaction strength) is captured by the *blockade radius*, defined as $R_b \equiv (\mathcal{V}_0/\Omega)^{1/6}$. The full phase diagram is thus parametrized by the ratio of the longitudinal to the transverse field Δ/Ω

and R_b/a , where a is the lattice spacing. For $\Delta/\Omega > 0$, the system energetically favors maximizing the number of atoms in the Rydberg state. However, this is subject to the blockade constraint, so for $R_b/a \gtrsim 1$, only one out of every pair of nearest neighbors can be excited; on a square lattice, this leads to the checkerboard phase with antiferromagnetic ordering of atoms in ground and Rydberg states. Higher values of R_b/a result in various new density-wave-ordered phases. Some of these correspond to classical hard-sphere packing of Rydberg excitations, while others are stabilized by quantum coherence between the ground and Rydberg states.

Recent experiments [5] have demonstrated three of these predicted states [Figs. 1(b)–1(d)], namely, the checkerboard, striated, and star phases. In the experiments, different values of R_b/a are accessed by tuning the lattice spacing a at fixed $\mathcal{V}_0$. By linearly ramping Δ/Ω from negative to positive values, one can quasi-adiabatically prepare different ordered states, which are probed by measuring targeted order parameters

constructed from the average of the Fourier transform intensities. However, structures incommensurate with the system size or the boundary conditions, and states with quantum fluctuations are challenging to detect and analyze directly from averages of Fourier transforms.

To address the challenge associated with large volumes of data produced by PQS projective measurements, we introduce an unsupervised-supervised hybrid machine learning approach: Hybrid-CCNN. In Ref. [11], some of us introduced the CCNN architecture, which modifies the standard convolutional network to reveal the key correlations of a learned phase. While the application of the CCNN to synthetic, classically simulated, data in Ref. [11] established the proof of principle, the supervised structure limited the CCNN's effectiveness. In particular, the CCNN trained on synthetic data failed to learn from experimental data. Moreover, even successful learning from synthetic data was tainted by the inherent bias of supervised learning associated with the choice of training sets. There have been efforts to remove such bias from "phase recognition" [12] algorithms either by using fully unsupervised learning [13–23] or by adopting an element of unsupervised learning [24–26]. However, discovery of new phases or fluctuation effects from experimental data through such efforts are rare to date. Building on the prior success in discovering new fluctuating phases from voluminous x-ray data using the Gaussian mixture model (GMM) [18], we introduce here the "hybrid-CCNN", which synergizes GMM-based unsupervised learning with improved CCNN-based supervised learning. We apply the hybrid-CCNN directly to voluminous Rydberg PQS data forgoing the use of synthetic data [see Fig. 1(f)]. Without requiring more experiments, the two-stage hybrid-CCNN yields three discoveries.

At each point in the (Δ, R_b) phase space, the available snapshot data consists of 250 binary maps $n_i(\mathbf{x}; \{\Delta, R_b\}) \in \{0, 1\}$, $i \in 1, \dots, 250$ with $\mathbf{x} \in [1, \dots, 13]^2$. An innovative feature of the hybrid-CCNN is the use of different data preprocessing for the unsupervised and supervised stages to target different facets of the high-dimensional data: average density modulations and snapshot-to-snapshot fluctuations. In targeting aspects of scientific importance, it is essential for human researchers to ensure that machine learning approaches do not latch onto trivial yet dominant effects. One such trivial effect is the overall density shift that occurs as the detuning Δ is increased. For the unsupervised learning stage of hybrid-CCNN, we enforce that the model ignores this behavior by constructing a set of Fourier-space features $|\tilde{n}(\mathbf{k})|^2$, which are invariant to the overall density shifts (see Appendix A).

In the final supervised stage, however, we work directly with position-space-based input from individual snapshots. This choice allows the hybrid-CCNN to capture the structure of many-body quantum fluctuations and learn from nonperiodic occurrences of motifs characteristic of a given phase. For this stage, we guide the CCNN away from the mean density profile by using the full density fluctuation maps $\delta n_i(\mathbf{x}) \equiv n_i(\mathbf{x}) - \bar{n}(\mathbf{x})$ produced from each snapshot as independent training data. Here, $\bar{n}(\mathbf{x}) = \sum_i n_i(\mathbf{x})/N$ is the snapshot-averaged Rydberg excitation density at each site $\mathbf{x}$,

where $N_s = 250$ is the number of snapshots at each (Δ, R_b) . As detailed later, this normalization allows the CCNN to easily focus on fully connected components of low-order correlation functions and prevents the network from learning trivial overall excitation densities.

II. UNSUPERVISED PHASE DISCOVERY

A. Feature selection and dimensional reduction

In scientific applications of ML, feature selection requires the most nontrivial involvement of human researchers. The chosen features should both reflect the scientific goal and enable separation of the trivially large signal from the scientifically interesting signal. At the same time, the feature choice should lead to a stable result that is robust to noise and insignificant details.

Since robust clustering typically requires a low-dimensional space, we form a representative feature vector for each point in parameter space by averaging over all snapshots measured at that point. Additionally, we desire these features to be independent from the trivial mean-density information $\bar{n}(\Delta, R_b)$. As we aim to discover spatially modulated patterns driven by the long-range interaction V , we choose our feature space to be in the Fourier basis, with the following preprocessing protocol. We start by normalizing the snapshots at each (Δ, R_b) by the mean density to obtain $\tilde{n}_i(\mathbf{x}) = n_i(\mathbf{x}) - \bar{n}$, where $\bar{n}$ is the average density across all sites and snapshots at (Δ, R_b) . Then, Fourier transforming to $|\tilde{n}_i(\mathbf{k})|^2$, subtracting the background contribution $\sim \sum_{\mathbf{k}} |\tilde{n}_i(\mathbf{k})|^2$, and averaging over all snapshots leads us to a density-shift-invariant structure factor.

To improve the robustness of the clustering, we further reduce the dimensionality of the feature space using principal component analysis (PCA) [27]. PCA identifies vectors called principal components (PCs) in this 256-dimensional feature space along which the data varies the most dramatically across the full phase diagram. Specifically, each of the principal components is a linear combination of multiple points in $\mathbf{k}$ space as visualized in Figs. 2(a)–2(c).

B. Clustering and results

The final step of the unsupervised learning stage is to cluster the phase-space points (Δ, R_b) in the reduced feature space spanned by the dominant PCA components. As our clustering algorithm, we use a Gaussian mixture model [27] (GMM) for its robustness and invariance to the scale of each feature. The two choices we must make are the number of principal components to keep for the clustering, and the number of clusters to fit. We choose the first by increasing the number of retained principal components one-by-one until the clusters stabilize, finding 10 to be sufficient for the current data. We then determine the optimal number of clusters to be six as the Bayesian information criterion [27] plateaus past this number (see Appendix A).

The clustering result shown in Fig. 1(f) partially resembles the manually obtained phase diagram based on evaluation of three target order parameters for the checkerboard, striated, and star phases [5] (see Appendix C), except that the unsupervised learning indicates that there are five phases distinct

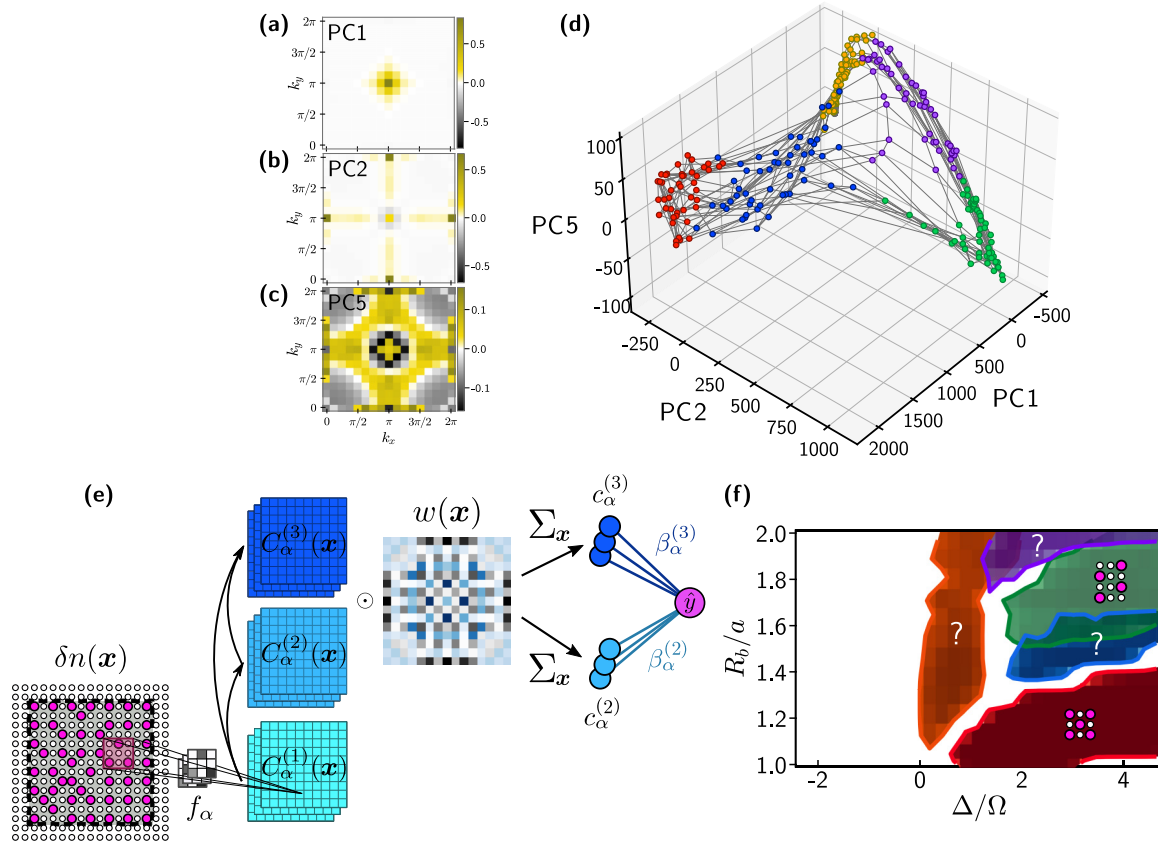


FIG. 2. [(a)–(c)] Principal components 1, 2, 5 in Fourier space. (d) Results of the GMM clustering performed in the reduced 10-dimensional PCA space and projected for visualization into the space spanned by (a)–(c). (e) The CCNN architecture for the supervised learning stage, constructed here up to third order $C_\alpha^{(3)}$ with three learned filters f_α and a learned spatial weighting $w(\mathbf{x})$. First, a density-normalized snapshot $\delta n(\mathbf{x})$ is convolved with the filters f_α to produce a convolutional map $C_\alpha^{(1)}(\mathbf{x})$. The $\delta n(\mathbf{x})$ are zero padded to allow a convolution of the filters over the entire snapshot. Then, a series of polynomials are applied to produce maps $C_\alpha^{(m)}(\mathbf{x})$, which measure m th-order correlators near each $\mathbf{x}$. These maps are summed with a learned spatial weighting $w(\mathbf{x})$ to produce features $c_\alpha^{(m)}$, which are used by a final logistic layer for classification. (f) Resulting phase diagram produced by supervised learning, obtained by cropping the classification confidence maps at level-set contours $\hat{y} = 0.75$ and overlaying them.

from the disordered phase [shown in grey in Fig. 1(f)]. A visualization of the clusters projected to the subspace spanned by PC1, PC2, and PC5 shows that the clusters are tightly defined [Fig. 2(d)], and notably the red, green, and purple clusters are located at high values of PC1, PC2, and PC5, respectively. As PC1 and PC2 show considerable overlap with the theoretical order parameters for the checkerboard and star phases, respectively, this allows us to clearly identify these phases with no prior knowledge. While PC3 and PC4 do not immediately offer interpretation as an order parameter in Fourier space, possibly capturing peak broadening (see Appendix A), PC5 somewhat resembles the Fourier-space order parameter associated with the rhombic phase predicted in DMRG simulations on a cylinder [9]. This is tantalizing since the previous manual analysis of the data did not report any signature of additional phases [5]. At this stage, two phases are clearly resolved: the checkerboard (red region and PC1) and the star phase (green region and PC1), we have hints at a potential rhombic-like phase (purple region and PC5), and the identities of the orange and blue phases remain unclear from the unsupervised learning alone.

III. SUPERVISED PHASE CHARACTERIZATION

A. Architecture and training

To better characterize each of the phases, we now turn to a supervised learning stage, which focuses on real-space snapshot-to-snapshot fluctuations $\delta n_i(\mathbf{x}; \Delta, R_b)$ [see Fig. 2(e) for the description of the architecture]. To learn the distinct identity of each phase, we train multiple neural networks, with each given the task of identifying snapshots of a single phase against the rest through a binary classification¹. The original CCNN architecture classifies phases by learning a set of spatially-averaged correlation functions parameterized using convolutional filters [11]. Specifically, from site-normalized density fluctuation snapshots $\delta n_i(\mathbf{x})$, the CCNN builds m th-order correlation maps $C_\alpha^{(m)}(\mathbf{x})$ parameterized by learnable

¹This is distinct from the common choice of using one neural network with multineuron output for multiphase detection [26,28,29].

filters $f_\alpha(\mathbf{a})$ through m th-order polynomials defined by

$$C_\alpha^{(m)}(\mathbf{x}) = \sum_{a_1 \neq \dots \neq a_m} \prod_{j=1}^m f_\alpha(\mathbf{a}_j) \delta n(\mathbf{x} + \mathbf{a}_j). \quad (2)$$

We evolved our supervised learning phase from that of Ref. [11] in three significant ways.

The first evolution is to introduce a learnable spatial weighting $w(\mathbf{x})$ to the CCNN [see Fig. 2(e)] that can reveal spatial structures beyond the length scale of the filters. The new spatially weighted features are

$$c_{\alpha, w(\mathbf{x})}^{(m)} \equiv \sum_{\mathbf{x}} w(\mathbf{x}) C_\alpha^{(m)}(\mathbf{x}). \quad (3)$$

We restrict $w(\mathbf{x})$ to be symmetric under reflections and rotations of the spatial coordinates for simplicity of parametrization. The second evolution is to restrict the filters to be positive definite, which aids interpretation by avoiding additional sign bookkeeping. The third evolution is to determine the minimal order m in a systematic manner through ablation testing [30] and strong regularization (described in Appendix A).

To produce the final classification, the spatially weighted features $c_{\alpha, w(\mathbf{x})}^{(m)}$ enter a final logistic layer coupled by weights $\beta_\alpha^{(m)}$ and bias ϵ to produce the output

$$\hat{y} = \left[1 + \exp \left(- \sum_{\alpha, m} \beta_\alpha^{(m)} c_{\alpha, w(\mathbf{x})}^{(m)} + \epsilon \right) \right]^{-1}. \quad (4)$$

For training, the snapshots from the target training points are labeled with $y = 1$ and those from the remaining training points are labeled with $y = 0$. During training, the filters $f_\alpha(\mathbf{a})$, logistic weights $\beta_\alpha^{(n)}$, bias ϵ , and spatial weighting $w(\mathbf{x})$ are all simultaneously learned by stochastic gradient descent to minimize the cross-entropy loss, which drives the predicted $\hat{y}$ s towards their correct labels y (Appendix A).

This architecture enables access to new theoretical insights by offering both expressibility through the nonlinear feature maps, and interpretability by ensuring these features are physically meaningful and linearly coupled to the output. Our three modifications in this paper further aid in this goal. Specifically, the learned spatial weighting $w(\mathbf{x})$ spotlights emergent heterogeneity unique to each phase. The positivity of the filter $f_\alpha(\mathbf{a})$ allows us to unambiguously infer the ferromagnetic (+) or antiferromagnetic (−) nature of the learned characteristic m -pixel correlations from the sign of the learned weights $\beta_\alpha^{(m)}$. Finally, a principled approach to the decision of the necessary order m of the correlation features provide confidence in the significance of the learned filter-specific m -pixel correlations. In the rest of this section, we present the theoretical insights gained in this manner.

B. Independent learning of known order parameters

We first focus on the red and green phase-space regions in Fig. 2(f). Comparing these two regions to the earlier result [5] based on manual evaluation of target order parameters in Fourier space, the red and green regions of Fig. 2(f) clearly map to the checkerboard [Fig. 1(b)] and star [Fig. 1(d)] phases, respectively. For these phases, the key advantages of CCNN-based phase recognition are an *unbiased* discovery of

the simplest order parameter suitable for the complexity of fluctuations present in the data (see Appendix B), as well as noticeably sharper phase boundaries without prior knowledge.

While these regions coincide with checkerboard and star phases manually identified previously, hybrid-CCNN proves itself by rediscovering the known order parameters through unbiased search. Specifically, the learned minimal architecture of uniform spatial weighting and two-point correlations for the red and green phases allows us to connect the learned features to the previously inspected Fourier-space-based order parameters. As shown in the Appendix C, by the convolution theorem, a linear combination of $c_{\alpha, w(\mathbf{x})=1}^{(2)}$ features amounts to a sum of the structure factor $|\delta n(\mathbf{k})|^2$ weighted by the Fourier transform of the filters $f_\alpha(\mathbf{a})$. Inspecting the learned effective weightings in Fourier space (shown in Appendix C), we remarkably find that the learned $c_{\alpha, w(\mathbf{x})=1}^{(2)}$ identify the correct order parameters traditionally used to characterize the respective density-wave orderings. We emphasize the nontriviality of the former observation as the hybrid-CCNN is able to reveal the correct order parameters *without* any prior input about the physics or structure of the density-wave-ordered ground states. This highlights the utility of our method for applications to potentially more complicated symmetry-breaking phases, for which the correct order parameters may not be immediately obvious.

C. Insights

In the blue, orange, and purple phases, the hybrid-CCNN provided insights from the data without requiring new experimental data. The discoveries were propelled by the synergy between the implemented spatial weight $w(\mathbf{x})$ learning and the CCNN's innate ability to learn key multipixel correlations.

1. Blue region: Fluctuations in the striated phase

The blue region found in the unsupervised stage overlaps with the region previously demarcated as the striated phase. The supervised phase of the hybrid-CCNN sharpens the phase boundary and offers evidence of nontrivial quantum many-body correlations in this region. To gain insight into the identity and refine the phase boundary of the blue phase, we locate the most informative and stable training points (see Appendix B) and use the resulting trained CCNN to mark the phase-space region supporting the phase by producing the average confidence map $\hat{y}(\Delta, R_b)$. The resulting phase boundary is remarkably sharper than previously detected using a mean-field order parameter. Moreover, the learned spatial weighting $w(\mathbf{x})$ and the learned logistic weights $\beta_1^{(2)}, \beta_2^{(2)}$ also encode new information beyond the mean field.

First, the weight map $w(\mathbf{x})$ learns to identify a specific sublattice in the bulk, and correspondingly activates the filters only when they are centered on this sublattice. As a result, the CCNN measures correlations within repeating 3×3 blocks that span the system [see Fig. 3(e)]. Second, the sign of the learned logistic weights $\beta_1^{(2)}$ and $\beta_2^{(2)}$ provides tomographic insight into the many-body state in this region. While the filter f_1 and its positive weight confirms joint excitation of corner sites in the 3×3 block expected in the striated phase, the filter f_2 and its negative weight [see Fig. 3(c)] reveals quantum cor-

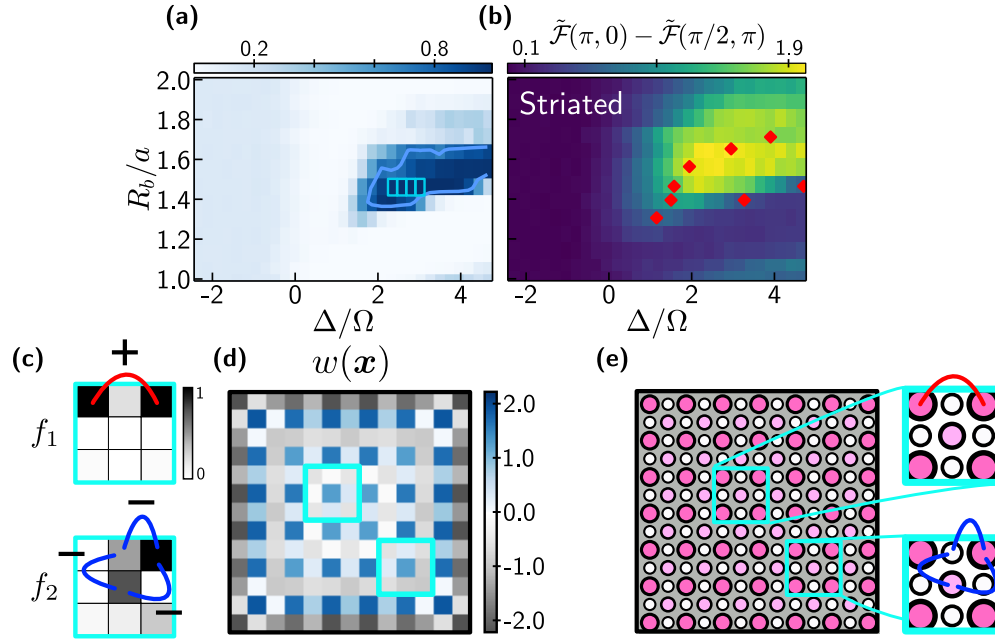


FIG. 3. (a) CCNN-learned region of support for the striated phase, with highlighted boxes indicating the training points. (b) Previously used approximate order parameter detecting the striated phase. Red markers indicate phase boundaries obtained from DMRG simulations on a 9×9 array [5]. (c) The filters learned by a third-order nonuniform CCNN to identify the striated phase in (a) and the signs on the $\beta_\alpha^{(2)}$ coefficients connecting the corresponding $c_\alpha^{(2)}$ to the output. For ease of display, the filter weights are normalized such that the maximum is 1 within each filter. (d) The spatial weighting $w(\mathbf{x})$ learned by a third-order CCNN identifying the striated phase. A single-pixel outer layer, corresponding to where the filter lands on the zero-padded region, is omitted for clarity. (e) A diagram showing example patches of the idealized striated phase whose correlations are measured by the CCNN of (c) and (d).

relations incompatible with a product-state ansatz. Moreover, anticorrelations among all three sites in the unit cell frustrates the polarization on each site.

The tomographic information advances our understanding of the role quantum dynamics play in Rydberg simulations as captured by the experimental data. Without the quantum fluctuation driven by the transverse field Ω , classical energetics will dictate one of the classical ordering patterns, as was pointed out in earlier DMRG simulations [9]. At the same time, our insights revealing connected quantum fluctuations present a conundrum. Firstly, the observed quantum correlations rule out the earlier product-state ansatz, which only captures superposition between $|r\rangle$ and $|g\rangle$ states on individual sites [5]. Secondly, the observed deviations in the PQS data from the mean-field product-state ansatz contradict DMRG findings reporting good overlap between the DMRG ground state and the mean-field ansatz for the striated phase [5,31].

Quantum correlations and entanglement can arise from two primary sources. Firstly, they might be present in the ground state itself, particularly in the vicinity of a second-order quantum phase transition [32]. Second, they might be generated in the dynamical state preparation process due to the quantum Kibble-Zurek mechanism [33], where nonadiabatic processes can coherently generate superpositions including excited states that generically result in entanglement. Our hybrid-CCNN approach cannot distinguish between these scenarios as it is agnostic to the actual origin of the correlations. Nonetheless, to better understand the potential entanglement structure, we inspect the bipartite entanglement entropy and

correlations within a 9×9 system using the density-matrix renormalization group (DMRG) in Appendix D. We find that the calculated von Neumann entanglement entropy S peaks sharply along transition lines, before plateauing to a small but nonzero value within the phase. This is accompanied by anticorrelations between the excited sublattices as found by the CCNN, although the state preparation process appears to significantly extend the support of these correlations as compared to the DMRG ground state (see Appendix D).

2. Orange region: The boundary-ordered phase

Next, we turn to the first of the two mysterious phases, depicted in dark orange in Fig. 4(a). The first clue regarding the identity of this phase comes from the learned weight map $w(\mathbf{x})$, which focuses strongly on the edges of the snapshots. As shown in Fig. 4(b), the CCNN learned to measure the differences in correlations between the bulk and the boundary by having large $w(\mathbf{x}) > 0$ along the edge and predominantly $w(\mathbf{x}) < 0$ in the interior. The learned filters focus the CCNN's attention on specific short-range two-point correlations that differ significantly between the edge and bulk. Figure 4(c) demonstrates the dramatic performance gain enabled by the learned edge-centered weight map. Inspection of the experimental snapshots in this orange phase [Fig. 4(d)] indeed confirms a regular occurrence of local $\mathbb{Z}_2$ patterns of $(\bullet \circ \bullet)$ along the edges of the snapshots. In contrast, the bulk of the snapshots appear disordered, further evidenced by explicit

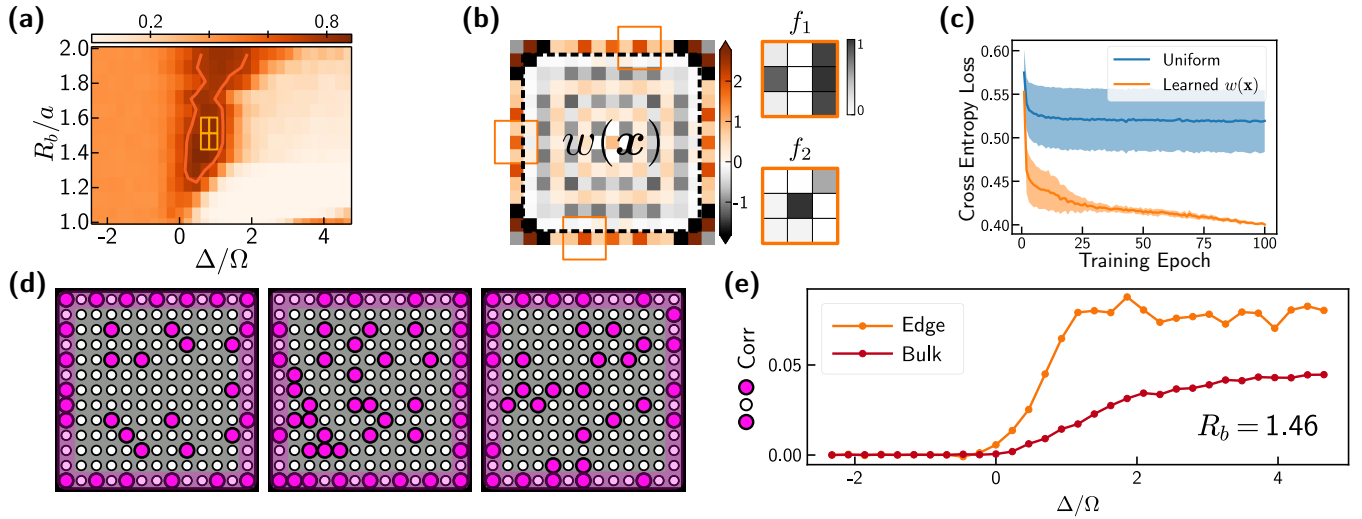


FIG. 4. (a) CCNN-learned region of support for the edge-ordered phase, with highlighted boxes indicating the training points. (b) The learned spatial weighting and filters for the model trained to identify the edge-ordered phase, with the spatial extent of the snapshot indicated by the dashed line. The outermost pixels correspond to where the filter is centered on zero padding but “clip” the edge pixels of the snapshot. For display purposes, the filter weights are normalized such that the maximum is 1 within each filter. (c) Measured performance discrepancy between second-order models with a fixed uniform $w(x) = 1$ and a freely learned spatial weighting. The central lines and bands show the mean and standard deviation across five randomly initialized models of each type, respectively. (d) Experimental snapshots from the training set, showing $\bullet \circ \bullet$ motifs primarily only along the single-site border, with the interior highly disordered. (e) Measurement of the third-nearest-neighbor $\langle \delta n_{i,j} \delta n_{i+2,j} \rangle$ connected correlation function within the edge and bulk (all sites but the outermost two-site strips) along a cut at $R_b = 1.46$, averaged across translations and other symmetries.

measurements of correlation functions along the edge and in the bulk in Fig. 4(e).

While the local $\mathbb{Z}_2$ pattern is commensurate with the neighboring checkerboard and striated phases, the reduced energetic cost of having Rydberg excitations along the boundary (relative to the bulk), due to the presence of fewer neighbors, actually allows the edge to order *before* the bulk. Hence, we identify this mystery phase as a boundary-ordered phase characterized by the edge ordering in the absence of long-range order in the bulk. The subsequent onset of bulk order, in the presence of preexisting edge order, defines an “extraordinary” boundary universality class [34]. We highlight the existence of this boundary-ordered phase in the present paper since this phase cannot be obtained on geometries with fully periodic or cylindrical boundary conditions, as was used for earlier DMRG calculations [9]. Critical to the identification of this phase is the real-space nature of the CCNN analysis as the edge ordering introduces a large number of artifacts into $\langle |n(\mathbf{k})|^2 \rangle$, which can challenge traditional Fourier-based analysis. Interestingly, a complementary paper [35] independently detected this edge ordering in quantum Monte Carlo simulations of the system with open boundary conditions, and confirmed the first-order nature of several transitions.

3. Purple region: Rhombic, staggered, and nematic motifs

Finally, we examine the other mystery phase identified by the hybrid-CCNN: the purple swath in Fig. 5(a). To identify the defining characteristic of this phase, we restrict the CCNN to learn positive correlation functions by enforcing $\beta_\alpha^{(n)} \geq 0$ during training, increase the filter size to 4×4 , and fix uniform $w(x) = 1$ (see Appendices C and D). The

CCNN learns the two filters f_1 and f_2 shown in Figs. 5(b) and 5(c) and uses a combination of two- and three-pixel correlations $c_{\alpha, w(x)=1}^{(2)}, c_{\alpha, w(x)=1}^{(3)}$ to recognize this phase as displayed in Figs. 5(d)–5(i). Inspection of the experimental data in Fig. 5(k) shows that these motifs are indeed present in the snapshots, albeit in short-ranged manners. The short-ranged nature of the occurrences indicate that the underlying tendencies for long-range ordered phases are being frustrated by the finite size of the system and the boundary conditions. Nevertheless, identification of characteristic motifs reveals the natures of the candidate states. Five of the six motifs indicate the presence of ferromagnetic and antiferromagnetic correlations (see Appendix E), which occur in the staggered and rhombic phases predicted to be present in this region of the phase space by Ref. [9]. Meanwhile, the final motif in Fig. 5(g) is expected to be common in the finite-size realization of the nematic phase proposed in recent DMRG simulations that retain the full long-ranged interaction [31].

The staggered and rhombic phases share the characteristic motifs in Figs. 5(k) and 5(l), but arrange these motifs differently. Earlier DMRG simulations with truncated interactions predicted the staggered phase at a larger R_b/a than this experimental range probes [9]. Recently, simulations retaining the full interaction range have found that in the presence of the long-ranged interaction tails, the rhombic phase is replaced in the phase diagram by a region of tight competition between a staggered order and an exotic nematic phase ORourke and Chan [31]. The nematic phase is a phase characterized by anisotropic modulation of the entanglement entropy. However, the exact nematic phase is vulnerable to finite-size boundary effects and Ref. [31] found a “nematic-like” state with the relevant entanglement properties in the bulk by im-

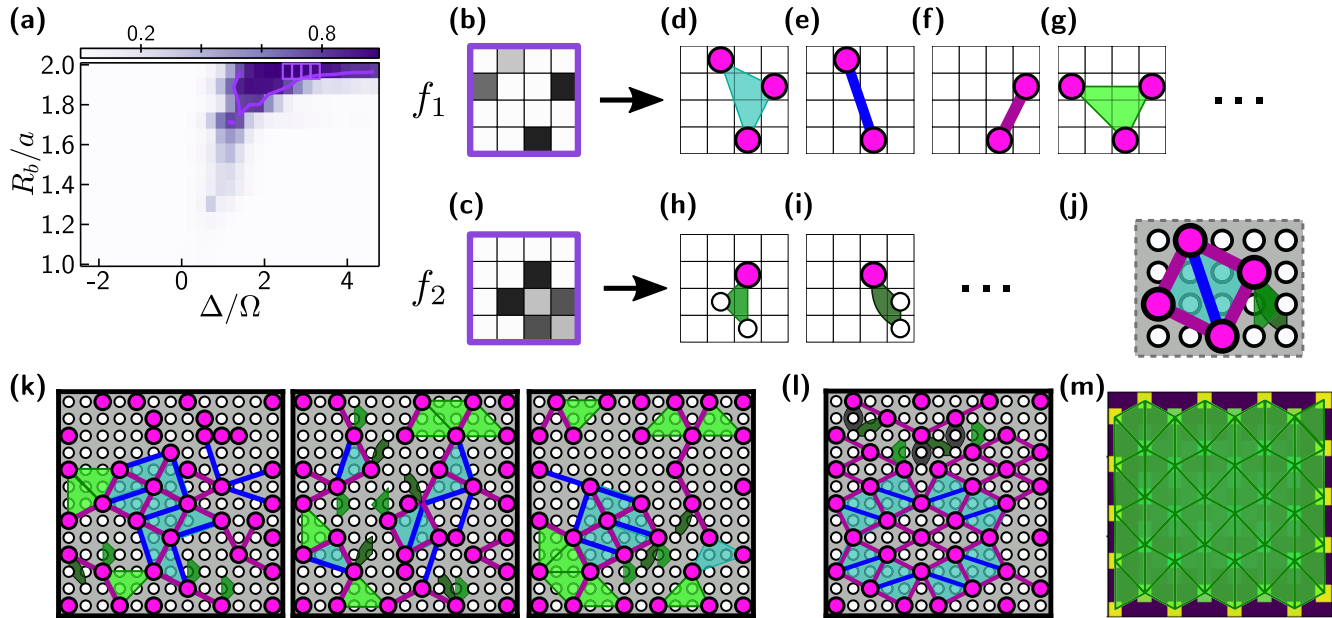


FIG. 5. (a) The region of support for the ordered phase(s) at large R_b/a , as learned by the full CCNN model of Fig. 1(n), with 5×5 filters and highlighted boxes indicating the training points. [(b),(c)] The two learned 4×4 convolutional filters for a simplified model ($w(\mathbf{x}) = 1$, $\beta_\alpha^{(n)} \geq 0$). [(d)–(i)] High-weight two- and three-point connected correlators measured by $c_\alpha^{(2)}$, $c_\alpha^{(3)}$ resulting from the filters in (a) and (b). We find $\beta_2^{(2)}$ to be nearly zero, so we omit two-point correlators stemming from the filter f_2 . Our CCNN is symmetrized (see Appendix B) and thus measures all correlators symmetry-equivalent under rotations and flips to those shown. (j) Identification of local occurrences of these motifs in experimental snapshots sampled from the training set. (k) Identification of five of these motifs in the ideal rhombic pattern. (l) Identification of five of these motifs as constructing the ideal staggered pattern. (m) Identification of the final motif (g) as the key pattern in finite-size realizations of the nematic phase (image reproduced from Ref. [31], with permission).

posing a tailored edge configuration. Hence, the fate of the rhombic order, the staggered order and the highly entangled nematic phase in an experimental setting, and whether such an entangled state can be realized, are open questions.

To assess which of these phases is dominant, we identify the unique unit-cell motifs characterizing each phase [shown in Figs. 5(j), 5(l), and 5(m)] in the experimental data. We find that in each snapshot, there are on average 13.3(3) cells matching the staggered motif while there are 20.4(4) cells matching the nematic motif. We do not find the full rhombic motif. From these counts, we see that nematic motif best represents the hybrid-CCNN discovered purple phase. This is evidence of the highly entangled nematic phase in experimental snapshots.

IV. DISCUSSION

In summary, we have developed a supervised-unsupervised hybrid machine learning approach, the hybrid-CCNN, to reveal collective quantum phenomena in voluminous quantum snapshot datasets and applied our approach to square-lattice Rydberg PQS data. The initial unsupervised stage used Fourier intensities $\langle |n(\mathbf{k})|^2 \rangle$ and clustering in a low-dimensional feature space obtained using PCA to reveal a rough initial phase diagram. This first pass reveals the number of phases to expect and informs the initial location of training points for the supervised stage. The phase diagram is then refined in the second supervised stage by training CCNNs to identify a small set of key correlations describing each phase. The CCNNs in this paper are evolved in three ways compared

to the original one of Ref. [11]: through introduction of a spatial weighting, positive definite filters, and ablation testing. These adaptations led to insights and uncovered the identities of each phase, using a principled minimal architecture. The identities thus revealed not only confirmed the previously known phases [5,9] but also resulted in insight into potential quantum entanglement structures in the striated phase and the discovery of previously undetected phases: the edge-ordered phase, and the motifs of staggered and nematic orders. In particular, observation of the nematic motifs—that are expected to be highly entangled—is a nontrivial confirmation of predictions from recent DMRG simulations [31]. We emphasize that these findings were made using machine learning tools trained entirely on experimental quantum simulator datasets.

The insights that we gained on the square-lattice Rydberg array using hybrid-CCNN have significant theoretical and experimental implications. Firstly, the observation of enhanced quantum correlations in the striated phase raises questions on the interplay between ground-state entanglement and correlations resulting from the quantum Kibble-Zurek mechanism [33]. Furthermore, hybrid-CCNN-aided detection of quantum correlations will be instrumental in future studies of exotic states that are becoming experimentally accessible [36]. Secondly, the discovery of the boundary-ordered phase raises theoretical questions on the nature and mechanism of the phase transitions. Independent observation of boundary ordering in quantum Monte Carlo (QMC) simulations [35] supports the detected boundary ordering as a property of the ground state. Finally, the signature of the short-ranged nematic ordering observed to coexist with motifs of staggered

or rhombic order is the experimental evidence of a highly entangled nematic ground state. This success is significant since all these phases were predicted for a narrow range of parameter space in the thermodynamic limit and were expected to be sensitive to the boundary conditions. Our success in detecting motifs of these intricate orders paves the way for detecting other rarer and more complex phases in future PQS experiments.

With the rapid progress towards probing more exotic quantum many-body phenomena using PQS [36–39], the need for new data-centric approaches to extracting insight from large volumes of quantum snapshot data will only grow. Here, we demonstrated that the hybrid-CCNN can not only meet this need but also enable the discovery and identification of new quantum states. The hybrid-CCNN’s ability to extract spatial structures of a quantum state at multiple length scales is particularly valuable given the limited spatial extent and incommensurate domains of phases produced by finite systems under quasiadiabatic state preparation [33].

ACKNOWLEDGMENTS

C.M. acknowledges funding support by the U.S. Department of Energy, Office of Science, Office of Advanced Scientific Computing Research, Department of Energy Computational Science Graduate Fellowship under Award No. DE-SC002034. K.W., M.G., and E.-A.K. acknowledge support by the National Science Foundation through Grant No. OAC-1934714. This research was supported in part by a New Frontier Grant from Cornell University’s College of Arts and Sciences. R.S. and S.S. acknowledge support by the U.S. Department of Energy, Grant No. DE-SC0019030. S.E., T.T.W., H.P., and M.D.L. acknowledge financial support from the Center for Ultracold Atoms, the National Science Foundation, the U.S. Department of Energy (Grant No. DE-SC0021013 & LBNL QSA Center), the Army Research Office, ARO MURI, an ESQ Discovery Grant, and the DARPA ONISQ Program. The authors acknowledge helpful discussions with Soonwon Choi, Marcin Kalinowski, and Roger Melko. We would also like to thank H. Levine, A. Keesling, G. Semeghini, A. Omran, and D. Bluvstein for the use of experimental data presented in this paper. The DMRG calculations in this paper were performed using the ITensor package [40] and were run on the FASRC Odyssey cluster supported by the FAS Division of Science Research Computing Group at Harvard University.

APPENDIX A: UNSUPERVISED LEARNING DETAILS

To perform our initial rough-estimate unsupervised phase discovery, we first collect features at each experimental parameter point (Δ, R_b) . For easy interpretability, we choose to work with simple features that represent all second-order fluctuations, but are blind to the overall Rydberg excitation density present in the snapshots. We find that blinding our entire machine learning pipeline to the overall density results in phase boundaries that are more closely aligned to meaningful changes in spatial orderings rather than simple increases in Rydberg excitation densities. More complex phase transitions would require higher-order correlations or nonlinear unsuper-

vised learning techniques, but we find the following approach to be sufficient for our data. To allow for direct comparison to previous experimental paper [5], we would like to work with the average squared Fourier amplitudes of the snapshots at each (Δ, R_b) . In order to make the process blind to overall density changes, we must take two steps.

First, in this section, we work with snapshots, which are overall density-normalized as $\delta n_i(\mathbf{x}) \equiv n_i(\mathbf{x}) - \langle n \rangle$, where the expectation value is computed over all sites of all snapshots available at the same (Δ, R_b) . Note that this is different than the per-site density normalization used in the supervised follow-up. In the supervised phase, we found that per-site density normalization was necessary to build interpretable order parameters for phases with subtle correlation structures such as the striated, rhombic, and nematic phases, which could be masked by average density modulations induced by the edge ordering. Per-site density normalization subtracts out this average density modulation, allowing connected correlators to be easily measured. However, while difficult to interpret, these very same average density modulations appear to be key to the success of our unsupervised phase, which is restricted to measuring raw Fourier amplitudes.

We then measure the average squared Fourier amplitudes of these normalized snapshots,

$$\delta \hat{n}(\mathbf{k}; \Delta, R_b) = \frac{1}{N(\Delta, R_b)} \sum_{n_i \in (\Delta, R_b)} \left| \sum_{\mathbf{x}} e^{-i\mathbf{k} \cdot \mathbf{x}} \delta n_i(\mathbf{x}) \right|^2, \quad (\text{A1})$$

where $N(\Delta, R_b)$ is the number of available snapshots at the parameter value (Δ, R_b) .

Our first density normalization does not make $\delta \hat{n}(\mathbf{k})$ invariant under an overall shift in the density of the $n(\mathbf{x})$ s, even away from $\mathbf{k} = \mathbf{0}$, due to the input data being sampled from a bosonic system with the values of $n(\mathbf{x})$ restricted to 0 or 1. In this case, the marginal distribution of each individual site’s density is a Bernoulli distribution whose variance is linked to its mean as $\langle \delta n(\mathbf{x})^2 \rangle = \langle n(\mathbf{x}) \rangle (1 - \langle n(\mathbf{x}) \rangle)$. In this way, information about the density can “bleed” through to the learning process at all $\mathbf{k}$ points. We can see this explicitly by expanding out the square in Eq. (A1) and rewriting it as

$$\delta \hat{n}(\mathbf{k}; \Delta, R_b) = \frac{1}{N(\Delta, R_b)} \sum_{n_i} \left[\sum_{\mathbf{x}} \delta n_i(\mathbf{x})^2 + \sum_{\mathbf{x} \neq \mathbf{x}'} e^{-i\mathbf{k} \cdot (\mathbf{x} - \mathbf{x}')} \delta n(\mathbf{x}) \delta n(\mathbf{x}') \right]. \quad (\text{A2})$$

To make our Fourier-space features invariant under density shifts we need to remove the first term. Using Plancherel’s theorem, we can achieve this by normalizing once more in $\mathbf{k}$ space as

$$\hat{p}(\mathbf{k}; \Delta, R_b) = \delta \hat{n}(\mathbf{k}; \Delta, R_b) - \frac{1}{L^2} \sum_{\mathbf{k}'} \delta \hat{n}(\mathbf{k}'; \Delta, R_b), \quad (\text{A3})$$

with L being the number of $\mathbf{k}$ points sampled along each direction of the discrete Fourier transform. These resulting features $\hat{p}(\mathbf{k}; \Delta, R_b)$ are then provided to the principal component analysis, the results of which are summarized in Fig. 6. Figure 6 shows the resulting top 12 principal components, which together capture >99.9% of the variance of the dataset. Notably,

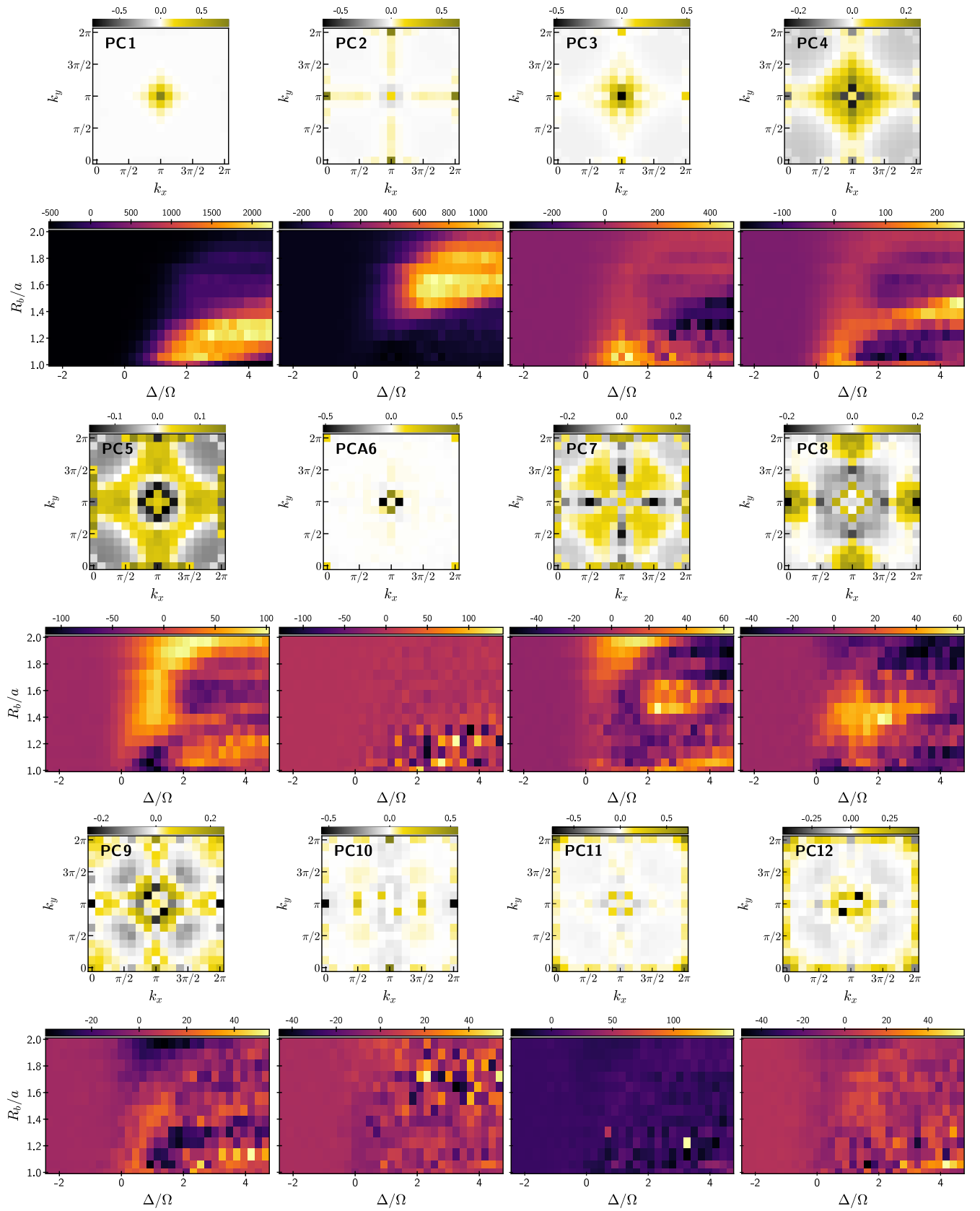


FIG. 6. The top twelve principal components, shown as weightings in k space, and the average projection onto each component across the experimental parameter space.

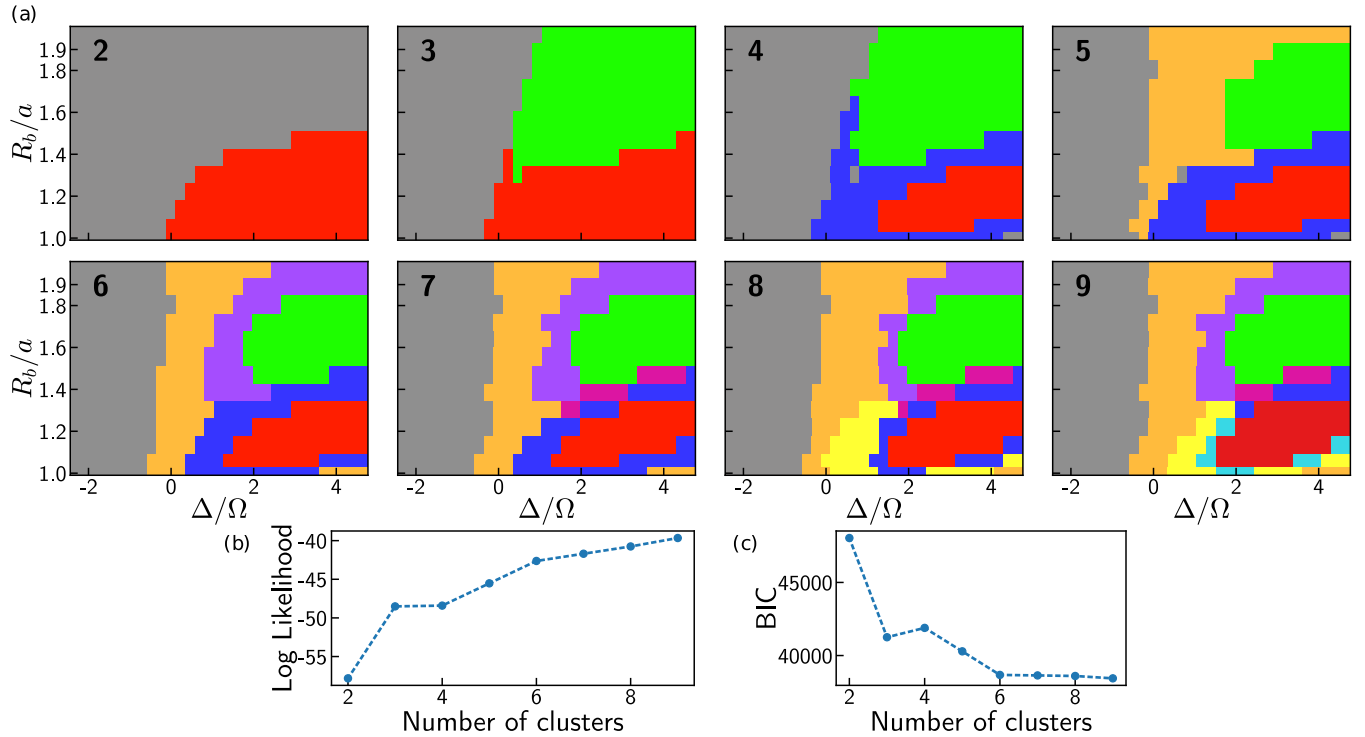


FIG. 7. (a) Clusters obtained when performing a Gaussian mixture model clustering with varying number of clusters (from 2–9), and the top 14 PCA components retained. Colors of each cluster are manually chosen for visual continuity. [(b),(c)] The log-likelihood and Bayesian information criterion (BIC) for the best mixture model at each number of clusters.

since all of the input features lie within the $\sum_k \hat{p}(\mathbf{k}) = 0$ subspace, so do the resulting principal component vectors.

Below each PCA vector in Fig. 6, we show its average inner product with each $\hat{p}(\mathbf{k}; \Delta, R_b)$ across parameter space. These maps reveal which areas of parameter space have average Fourier-space intensity patterns that match the principal component vectors well. Importantly, many of these maps vary smoothly in parameter space and as such, correspond to meaningful changes in Fourier structure. Meanwhile, at first glance PCA6, and partially PCA10, seem visually noisy in parameter space. However, this is due to these PCA components breaking rotational symmetry in Fourier space. In phase regions where the relevant Fourier peaks are present, these PCA components are either strongly positive or strongly negative depending on which way the symmetry is broken, while in other regions they remain close to zero. These components could be improved by further post-processing; however, for simplicity, we do not do so here.

To robustly perform the GMM clustering, we initialize the clusters using the k-means algorithm [27], repeat the initialization and clustering with different random seeds until 500 sequential clusterings do not improve the final log-likelihood, and keep the best-found clustering. Interestingly, random initializations (rather than k means) often produce clusterings with higher final log-likelihoods but poor structures in parameter space, with one cluster often a seemingly random collection of points across parameter space. This ambiguity points to the persisting need to have a physicist “in the loop” verifying machine learning (ML) results at each stage, which is made easier when using interpretable techniques.

To determine the appropriate number of clusters and PCA components to retain, we perform the entire clustering process while varying the number of clusters and the number of retained PCA components. In Fig. 7(a), we find that starting from two clusters, each additional cluster meaningfully captures the next-strongest phase in the dataset, with topology similar to the clustering of the main text with $N_{\text{clust}} = 6$. As expected, each additional cluster improves the final log-likelihood [Fig. 7(b)]; however, a distinct kink and reduction in slope is observed after $N_{\text{clust}} = 6$. This is reflected additionally by the Bayesian information criterion (BIC) [27], a standard heuristic metric for determining the correct number of clusters, plateauing past this point. As we should pick the simplest model, which minimizes the BIC, this points to $N_{\text{clust}} = 6$ as being optimal. Clusterings with $N_{\text{clust}} > 7$ are increasingly noisy and can be seen to simply be chunking off transition regions between phases, rather than dramatically changing the phase diagram’s topology.

In Fig. 8, we show clustering results with a fixed $N_{\text{clust}} = 6$ but varying number of PCA components kept. Across all settings, we predominately see the topology presented in the main text (similar to $N_{\text{PCA}} = 9$), with exceptions being $N_{\text{PCA}} = 5, 6, 7$, which substitute the purple phase with a cluster around the broadened-checkerboard region characterized by PCA3. The relative stability and understandability of these clusterings give us confidence that they indicate real regions of varying orderings suitable for a follow-up analysis. As truncating the number of PCA components kept is primarily a cost-saving and robustness-improving technique, we increase the number of principal components kept until the cluster-

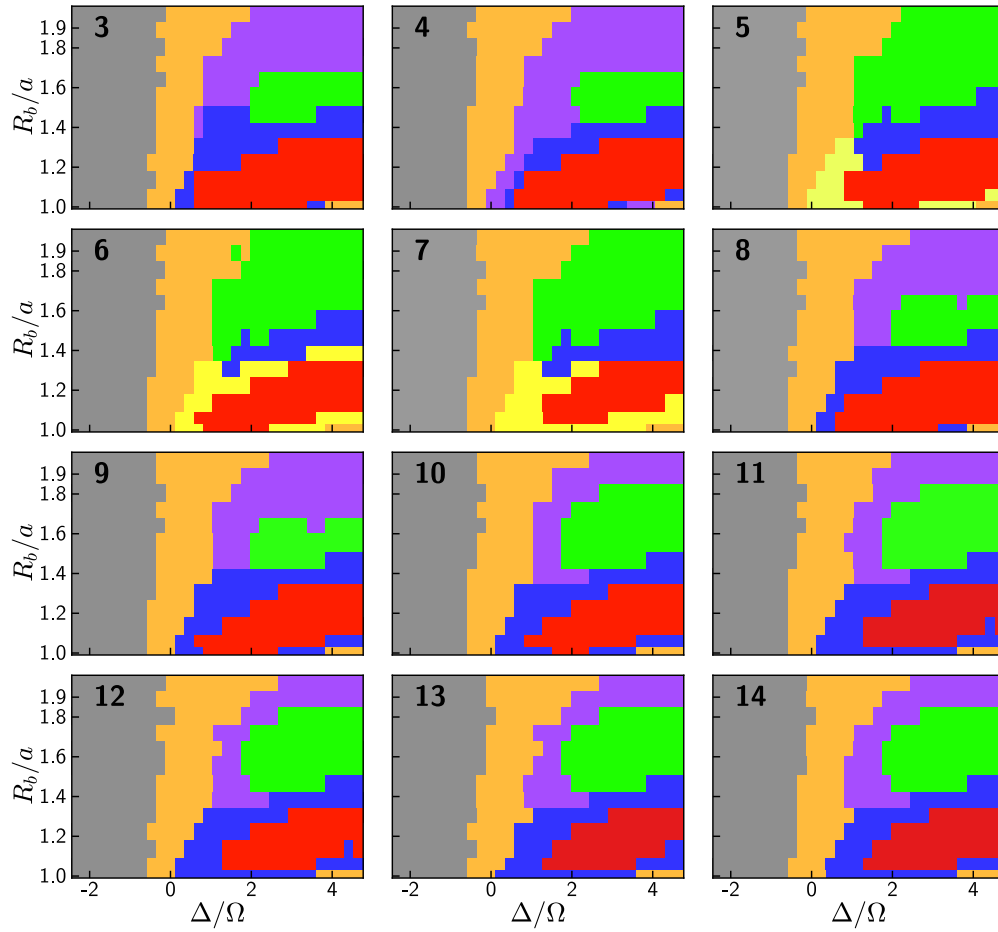


FIG. 8. Clusters obtained when performing a Gaussian mixture model clustering with six clusters, and varying the number of PCA components retained (from 3–14). Colors of each cluster are manually chosen for visual continuity.

ing is seen to stabilize, which can be seen in Fig. 8 to be roughly 10.

APPENDIX B: SUPERVISED LEARNING DETAILS

1. CCNN Training

We implement our Convolutional Correlator Neural Networks using the Pytorch [41] library, and our code is made available at GitHub [42]. Each CCNN is trained to identify a single phase: all snapshots sampled within that phase are labeled 0, while snapshots sampled from all other phases are labeled 1. The list of parameter-space points sampled to form each phase’s training set is given in Table I. From each point, we randomly take 90% of the snapshots as the training dataset shown to the network, and take the remaining 10% as a validation dataset, which is not shown during training and is only used to verify that the network has not overfit. To handle the uneven distribution of snapshots available from each phase, when training a CCNN to identify a phase $\mathcal{P}$ each presented snapshot has a 50% probability to be sampled from phase $\mathcal{P}$, and a 10% probability to be sampled from each of the 5 other phases. This ensures that there is an equal representation of “within-phase- $\mathcal{P}$ ” snapshots and “out-of-phase- $\mathcal{P}$ ” snapshots, as well as equal representation among all classes within the “out-of-phase” distribution.

The training points in Table I are obtained by starting with points suggested by the unsupervised clustering, and modifying iteratively based on the results of training until all phases overlap minimally and have clear distinctions. For the striated phase, the support of which was very narrow in the unsupervised learning results, we explored a wider range in phase space to identify the training points yielding physically meaningful features.

During training, the free parameters of our CCNNs $\{f_\alpha(\mathbf{a}), \beta_\alpha^{(m)}, \epsilon, w(\mathbf{x})\}$ are all simultaneously learned so as to minimize the training loss averaged across the dataset, chosen to be the standard cross entropy loss used for classification tasks with

TABLE I. Parameter-space points used to train models identifying each phase. Training sets are formed by all points in the Cartesian product of the Δ/Ω and R_b/a columns.

Phase	Δ/Ω	R_b/a
Checkerboard	3.02, 3.26	1.13, 1.23
Striated	2.33, 2.56, 2.79, 3.02	1.46
Star	3.95, 4.19, 4.42, 4.65	1.71
Purple	2.32, 2.56, 2.79, 3.02	1.97
Edge	0.69, 0.93	1.46, 1.56
Disordered	−2.09, −1.62, −1.16, −0.4	1.13, 1.46, 1.81

an additional L1 regularization on the filter weights,

$$\mathcal{L} = \frac{1}{N} \sum_i (-y_i \log \hat{y}_i - (1 - y_i) \log(1 - \hat{y}_i)) + \gamma \sum_{\alpha, \mathbf{a}} |f_{\alpha}(\mathbf{a})|, \quad (\text{B1})$$

where i runs over all snapshots in the training dataset, and N is the total number of snapshots. We optimize this loss using the ADAM [43] optimization algorithm, with a minibatch size of 128 snapshots, an initial learning rate of 0.01, and a cosine-annealed learning rate schedule as implemented by Pytorch's CosineAnnealingLR [41].

As in Ref. [11], we also place a BatchNorm [44] layer (without the optional affine transformation) after the correlator features $c_{\alpha}^{(m)}$, which introduces no extra free parameters but aids in rapid and stable convergence of the training. Similar to Ref. [11], we additionally spatially symmetrize $C_{\alpha}^{(m)}(\mathbf{x})$ and $w(\mathbf{x})$ to improve generalization and interpretability. Specifically, on each forward pass, we symmetrize the maps by summing over all symmetry transformations as

$$C_{\alpha}^{(m)}(\mathbf{x}) \leftarrow \sum_{g \in D_4} g C_{\alpha}^{(m)}(\mathbf{x}), \quad (\text{B2})$$

with the sum running over all group elements $g \in D_4$ acting on the convolutional maps, and the same transformation additionally applied to $w(\mathbf{x})$. This symmetrization procedure aids in generalization, as the model's predictions are made invariant under symmetry transformations of the input snapshots, and reduces the effective total number of parameters to learn. In particular, this allows the model to avoid having to learn symmetry-equivalent versions of the filters $f_{\alpha}(\mathbf{a})$, and makes the spatial weighting $w(\mathbf{x})$ easier to visually interpret. To make interpretation simpler, we also restrict $f_{\alpha}(\mathbf{a})$ to take only positive values by applying an absolute value function on every forward pass.

To ensure that all pixels of the learned filters hit all pixels of the snapshot in the convolution, we zero-pad each $\delta n_i(\mathbf{x})$ with a sufficient number of zeros. In particular, if the convolutional filters $f_{\alpha}(\mathbf{a})$ are of spatial extent $F \times F$, we pad the input snapshots with $F - 1$ zeros on all edges.

After training is completed, a CCNN has learned a collection of convolutional filters $f_{\alpha}(\mathbf{a})$, as well as a set of coefficients $\beta_{\alpha}^{(m)}$ connecting the m th-order feature $c_{\alpha}^{(m)}$ derived from filter f_{α} to the output, and an overall bias ϵ . The output of the model is then

$$\hat{y} = \left[1 + \exp \left(- \sum_{\alpha, m} \beta_{\alpha}^{(m)} c_{\alpha}^{(m)} + \epsilon \right) \right]^{-1}, \quad (\text{B3})$$

where $c_{\alpha}^{(m)}$ are constructed from the learned filters f_{α} and the input by Eq. (3) of the main text.

2. Ablation testing

When building a specific CCNN, there is a lot of flexibility in the architectural choices. These architectural hyperparameters include the order m to truncate at, whether to include a learnable spatial weighting $w(\mathbf{x})$, and how many filters (and of what size) to use. For the current paper, we take the approach

of building the simplest architecture [second-order, uniform $w(\mathbf{x})$] first, and adding on architectural complexity piece-by-piece. If a large gain in accuracy is achieved by a single architectural addition, then we keep that piece and attribute a quality of the phase to requiring the additional expressibility. This approach is commonly referred to as *ablation testing* in the ML literature [30], although for large neural networks, ML practitioners commonly *remove* modules piece-by-piece rather than adding them as we do for our shallow networks.

To improve interpretability, an additional hyperparameter that we have at our disposal is the coefficient γ on the filter L1 loss in Eq. (B1). Intuitively, larger γ results in simpler filters with more pixels deactivated but decreased classification performance. For all models except for the uniform second-order models (which we can easily interpret in Fourier space, see Appendix C), we increase γ until the filters identifying all phases are sparse enough to easily interpret while performance is maintained sufficiently high.

Along these lines, in Table II, we summarize validation accuracy measurements of several variations of our CCNNs trained to identify each phase. From these measurements, in conjunction with observing the quality of the resulting phase diagram, we can determine what is required to form a good order parameter for each of the identified phases. For example, the checkerboard, star, and purple phases show roughly uniform or even decreasing (due to overfitting) validation performance as the CCNN's truncation order is increased, indicating that second-order features are sufficient to distinguish these from other phases. However, we find the second-order features identifying the purple phase to be somewhat uninteresting as they primarily just measure the tendency for longer-range density correlations; see Appendix C. The star phase shows some improvement when incorporating spatial inhomogeneity, but we find that the phase diagram changes little while the second-order model is simplest to interpret (see Appendix C).

The most striking changes are observed for the two remaining phases, both of which benefit heavily from learning a nonuniform spatial weighting $w(\mathbf{x})$. For the striated phase, we observe a dramatic 11.6% jump in classification accuracy between uniform and nonuniform second-order models. This difference reflects that due to finite-size effects, many striated-like correlations persist throughout the system within the star phase. Appendix C presents an alternate interpretation in Fourier space, where the difficulty originates from the relevant Fourier peaks being overly diffuse.

However, many of these striated-like correlations in the star phase occur on the even sublattice, while the striated ordering entirely occupies the odd sublattice. If the network has the ability to focus on a specific sublattice matching the ideal striated ordering (making all other sites contribute with a negative weight to the output), it can better distinguish the striated from the star phase. Alternatively, we find that increasing the order of the model while keeping $w(\mathbf{x}) = 1$ also results in a good classifier for the striated phase but one which is more difficult to understand.

In contrast, for the edge-ordered phase, we find that simply increasing the order of the model while keeping uniform $w(\mathbf{x}) = 1$ makes negative changes to generalization

TABLE II. Final validation accuracies for models identifying snapshots of each phase using various architectures. Measurements are made with 10-fold cross-validation [27], with 5 random seeds run at each train/validation split. Reported errors for each model type are the standard error across all runs. All models are trained with 3 convolutional filters. Figure 2(f) of the main text is produced by the final, most expressive, row of models. Table entries in italics are simpler models studied in Figs. 4 and 5 of the main text. Models sampled from the “2nd order, $w(\mathbf{x}) = 1$ ” row produce the order parameters shown in Fig. 10 of Appendix C.

	Checkerboard	Star	Striated	Purple	Edge
2nd order, $w(\mathbf{x}) = 1$, 4×4 filters, $\gamma = 0.0$	99.81(1)	84.9(3)	75.7(3)	89.3(3)	78.0(3)
3rd order, $w(\mathbf{x}) = 1$, 4×4 filters, $\gamma = 0.1$	98.5(2)	73(1)	86.1(7)	87(1)	71.4(9)
3rd order, $\beta_{\alpha}^{(m)} \geq 0$, $w(\mathbf{x}) = 1$, 4×4 filters, $\gamma = 0.1$	98.7(2)	78(1)	83.3(5)	86.3(4)	71.6(8)
2nd order, learned $w(\mathbf{x})$, 3×3 filters, $\gamma = 0.1$	98.4(3)	83.6(9)	87.3(5)	88.5(4)	82.9(3)
3rd order, learned $w(\mathbf{x})$, 3×3 filters, $\gamma = 0.1$	100.	85.8(5)	91.9(2)	91.7(4)	88.0(2)

performance. Meanwhile, keeping the model at second order, we see a 4.9% jump in accuracy when allowing for a spatially varying $w(\mathbf{x})$. While this may not seem dramatic, we find that all models with uniform $w(\mathbf{x})$ produce order parameters, which persist deep into the disordered region. This indicates that successfully identifying this phase *requires* measuring spatially-inhomogeneous correlation functions, which reflects that this phase is itself *defined* by this inhomogeneity.

APPENDIX C: EXTRACTING FOURIER-SPACE ORDER PARAMETERS FROM UNIFORM SECOND-ORDER CCNNs

Second-order features with the uniform spatial weighting $w(\mathbf{x}) = 1$, $c_{\alpha, w(\mathbf{x})=1}^{(2)}$ amount to weighted sums of the Fourier spectrum of the input. Specifically, each feature $c_{\alpha, w(\mathbf{x})=1}^{(2)}$ can be written as

$$\begin{aligned}
 c_{\alpha, w(\mathbf{x})=1}^{(2)} &= \sum_{\mathbf{x}} C_{\alpha}^{(2)}(\mathbf{x}) \equiv \sum_{\mathbf{x}} \left[\left(\sum_{\mathbf{a}} f(\mathbf{a}) \delta n(\mathbf{x} + \mathbf{a}) \right)^2 \right. \\
 &\quad \left. - \sum_{\mathbf{a}} f(\mathbf{a})^2 \delta n(\mathbf{x} + \mathbf{a})^2 \right] \quad (\text{C1}) \\
 &= \frac{1}{N^2} \left[\sum_{\mathbf{k}} \left(|\hat{f}(\mathbf{k})|^2 - \frac{1}{N^2} \sum_{\mathbf{k}'} |\hat{f}(\mathbf{k}')|^2 \right) |\delta \hat{n}(\mathbf{k})|^2 \right], \quad (\text{C2})
 \end{aligned}$$

where Eq. (C1) is just the expanded definition of $C^{(2)}(\mathbf{x})$, Eq. (C2) is the Fourier-equivalent form, and discrete Fourier transforms are defined by

$$\hat{f}(\mathbf{k}) \equiv \sum_{\mathbf{a}=0}^{(L_f, L_f)} e^{-i\mathbf{k} \cdot \mathbf{a}} f(\mathbf{a}), \quad \delta \hat{n}(\mathbf{k}) \equiv \sum_{\mathbf{x}=0}^{(L_n, L_n)} e^{-i\mathbf{k} \cdot \mathbf{x}} \delta n(\mathbf{x}), \quad (\text{C3})$$

where L_f, L_n are the lengths of each dimension of the convolutional filter f and the input snapshot δn , respectively. The wavevectors in the discrete Fourier transform are defined as $k_i \in \{0, 2\pi/N, 4\pi/N, \dots, (N-1)2\pi/N\}$ with $N \geq L$ defining the resolution of the Fourier transform. Choosing $N > L$ encodes no new information in the result, but produces higher-resolution Fourier spectra for plotting.

Note that for all terms in Eq. (C1) to be well defined, we take an “infinite zero-padding” convention, where the domain of δn is expanded to all $\mathbf{x}$ by defining $\delta n(\mathbf{x}) = 0 \forall \mathbf{x} \notin$

$[0, L_n] \times [0, L_n]$. Under this convention, we take the $\mathbf{x}$ sum to be over all space. In practice, our CCNNs equivalently pad δn with enough zeros to recover all nonzero $C_{\alpha}^{(2)}(\mathbf{x})$. Equation (C2) can be obtained from Eq. (C1) by straightforward application of the convolution theorem and Plancherel’s theorem.

A simple interpretation of this result is that a uniform-weighted $c_{\alpha}^{(2)}$ measures a weighted sum of $\delta n(\mathbf{k})$, with the weights given by the Fourier transform of f_{α} , normalized to be zero mean in $\mathbf{k}$ space. Since second-order CCNNs linearly combine multiple filters f_{α} with coefficients β_{α} to produce the input to the final logistic function, we can understand the full action of the network by the weighted sum of the input in $\mathbf{k}$ space with effective weights $\tilde{f}(\mathbf{k}) = \sum_{\alpha} \beta_{\alpha}^{(2)} \tilde{f}_{\alpha}(\mathbf{k})$, with normalized Fourier intensities of each filter defined as $\tilde{f}_{\alpha}(\mathbf{k}) = |\hat{f}_{\alpha}(\mathbf{k})|^2 - \frac{1}{N^2} \sum_{\mathbf{k}'} |\hat{f}_{\alpha}(\mathbf{k}')|^2$. Due to the symmetrization procedure outlined in Appendix B, this map must be then symmetrized over all symmetries of D_4 . This “order parameter map” defines our effective second-order CCNN order parameter for a given phase as

$$O = \sigma \left(\sum_{\mathbf{k}} \tilde{f}(\mathbf{k})^{\text{sym}} |\delta \hat{n}(\mathbf{k})|^2 - \epsilon \right), \quad (\text{C4})$$

with $\sigma(x) = (1 + \exp(-x))^{-1}$ the logistic sigmoid function, and ϵ the learned bias.

In Fig. 9(a), we provide a visual diagram demonstrating this process to produce an interpretable $\tilde{f}(\mathbf{k})^{\text{sym}}$. In Figs. 9(b)–9(d), we show exemplary resulting Fourier-weighting maps learned for each of the checkerboard, striated, and star phases, and demonstrate the intuition to interpret these maps. Applying the weighting $\tilde{f}(\mathbf{k})^{\text{sym}}$ to the average Fourier intensity of the target phase should produce a large positive number, while applying it to the intensities of all other phases should produce smaller or negative numbers (after subtracting the learned ϵ , only the target phase should remain positive).

We find that the checkerboard, striated, and star Fourier weightings look strikingly similar to smeared-out versions of the hand-crafted order parameters discussed in Refs. [5,9], which can be interpreted in our framework as $\tilde{f}(\mathbf{k})$ that are nonzero only at finitely many $\mathbf{k}$ points. In particular, our learned checkerboard order parameter is strongly positive at (π, π) , the striated at $(\pm\pi, 0)$, $(0, \pm\pi)$, and the star order parameter spreads positive weight towards $(\pi/2, 0)$, $(0, \pi/2)$ while putting negative weight at (π, π) . This observation is

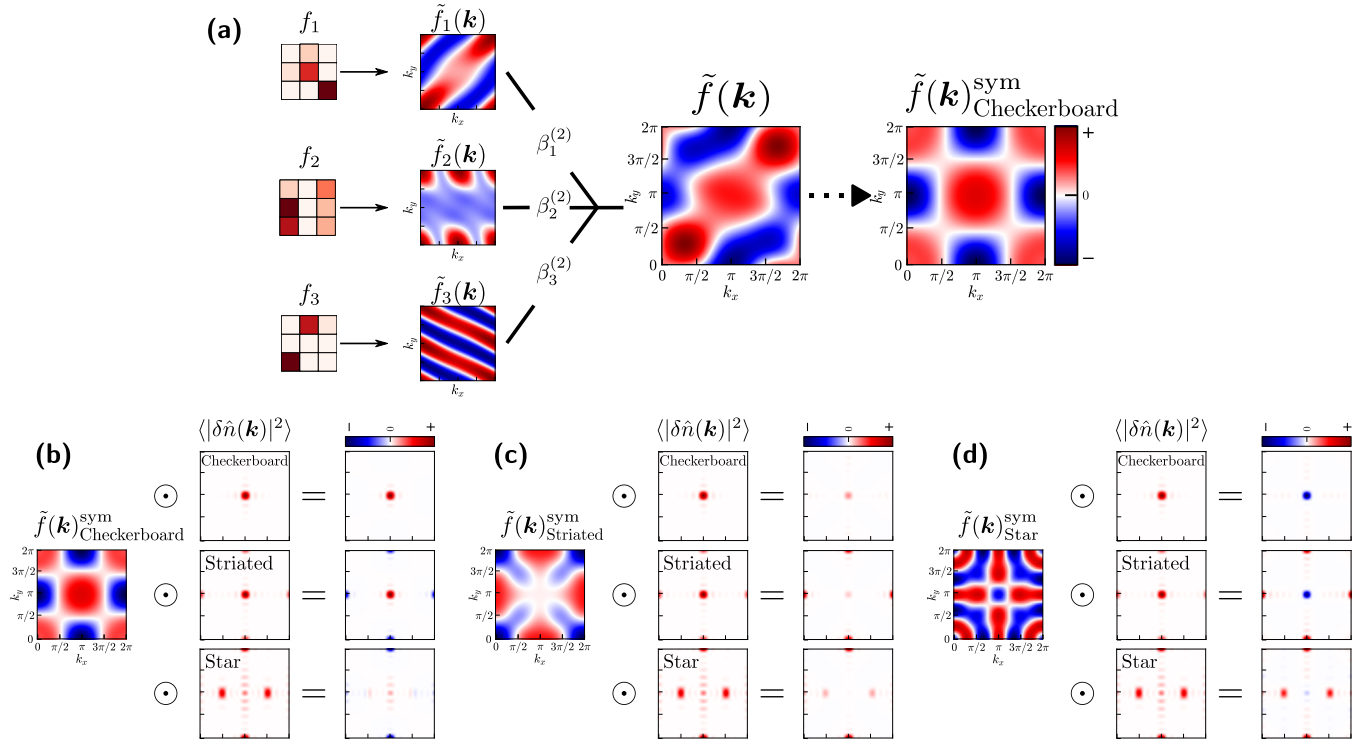


FIG. 9. (a) The process by which the action of a uniform- $w(\mathbf{x})$ second-order CCNN can be interpreted in Fourier space. Each filter is discrete Fourier transformed and normalized to form $\tilde{f}_\alpha(\mathbf{k})$, linearly combined pointwise in $\mathbf{k}$ space with learned coefficients $\beta_\alpha^{(2)}$, and then symmetrized to form the final map. [(b)–(d)] The order parameter maps $\tilde{f}(\mathbf{k})^{\text{sym}}$ learned to identify the checkerboard, striated, and star phases. We apply each order parameter map as weightings to the idealized Fourier intensities for each of the checkerboard, striated, and star phases. If learning is successful, applying this weighting and then summing in $\mathbf{k}$ space should produce large positive numbers for the target phase, and small or negative numbers for every other phase.

reassuring as it demonstrates that these models identify each phase by an ordering similar to the ideal density waves.

In Figs. 10(a)–10(c), we show measurements of Fourier-space order parameters manually crafted to capture each of the checkerboard, star, and striated phases. We can see that due to the blurring of the relevant Fourier peaks resulting from the finite-size system, the order parameters for the star and striated phases [Figs. 10(b) and 10(c)] heavily overlap. For comparison, in Figs. 10(d)–10(h), we show confidence maps from uniform second-order CCNNs with 4×4 convolutional filters trained to recognize each phase examined in the main text. We can observe that only the checkerboard (red) and star (green) phases are well resolved by these reduced models, in accordance with the intuition that these phases are classical crystals easily identified in Fourier space. Meanwhile, just as with manually crafted Fourier-space order parameters, the uniform second-order CCNN’s predictions for the striated phase significantly overlap with those of the star phase. These combined observations point to a fundamental limit to phase resolution using Fourier-space order parameters in finite, open-boundary, noisy systems. The purple phase is somewhat well resolved, but it too suffers from overly broad peaks in Fourier space. Meanwhile, we see that uniform second-order models fail dramatically for the edge-ordered (orange) phase, which by its nature fundamentally requires the ability to learn spatially inhomogeneous functions, as measured in Appendix B.

To uncover what is being learned by the second-order CCNN for the purple phase in particular, we repeat the above analysis to produce the Fourier-space order parameter shown in Fig. 11(a). We see that the learned order parameter attempts to identify diffuse Fourier intensity along the diagonals in Fourier space. In contrast, Fig. 11(b) shows that Fourier peaks appear at $(\pm 2\pi/5, 0)$ and $(0, \pm\pi/2)$ in the ideally-ordered rhombic phase. However, due to broadening resulting from experimental noise and the finite-size system, we hypothesize that the CCNN does not attempt to directly measure any long-range ordering peaks as they blur too strongly into the star phase’s $(\pm\pi/2, 0)$ peaks. Nevertheless, peaks resembling the ideal rhombic ordering can be visually resolved when averaging over a large number of snapshots as in Fig. 11(f).

Manual inspection of the learned filters and $\beta_\alpha^{(2)}$ coefficients reveals that the purple-phase CCNN obtains this order parameter by placing negative $\beta_\alpha^{(2)}$ on filters, which contain short-range patterns. This makes intuitive sense, as density fluctuations between nearby sites are anticorrelated at high R_b . However, this does not give us clear insight into the actual Rydberg crystal being realized, other than that it is constructed from longer-range displacements. This motivated our choice in the main text of investigating third-order models, which enforce $\beta_\alpha^{(m)} > 0$ to ensure that the CCNN must rely on positive correlations to identify the phase. Due to three-point correlators having a nontrivial sign structure (see Appendix E), the CCNN still measures some short-range patterns, but also

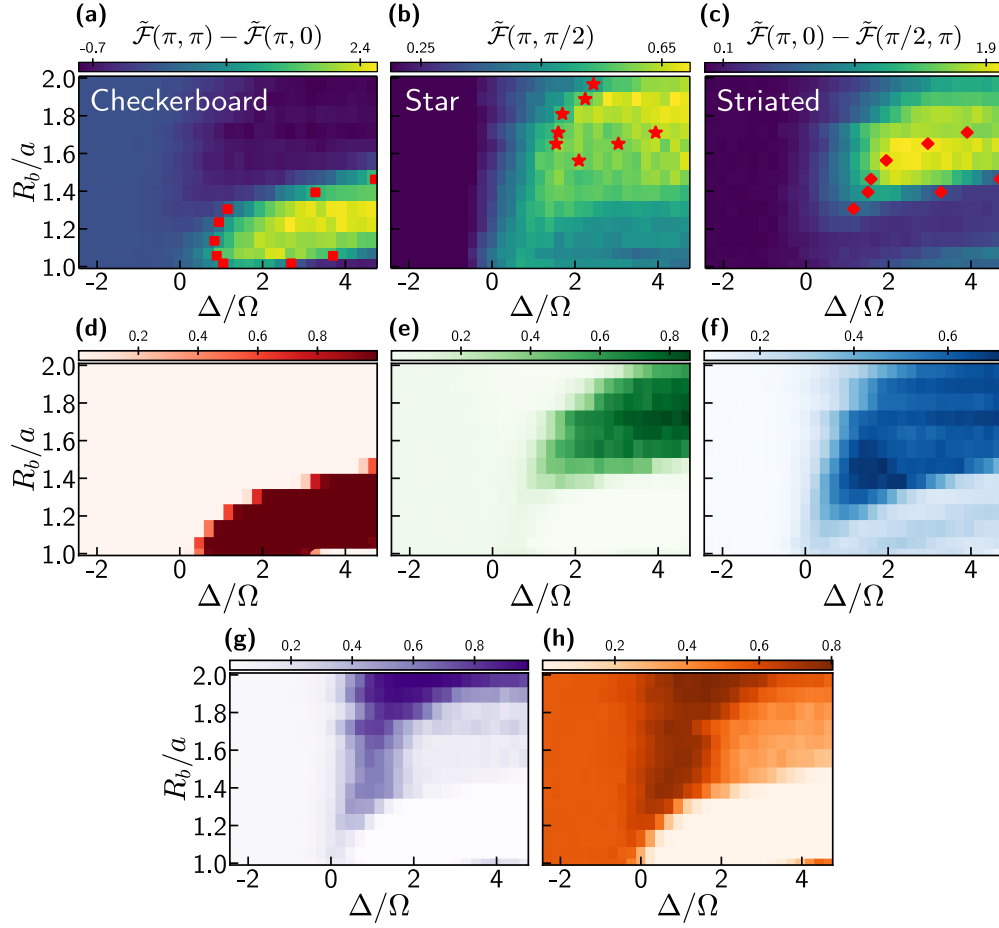


FIG. 10. [(a)–(c)] Measurements of Fourier-space order parameters manually crafted [5] to capture the checkerboard, star, and striated phases. Red symbols mark phase boundaries from DMRG simulations of a 9×9 open boundary system [5]. [(d)–(h)] CCNN order parameter maps of uniform second-order CCNNs trained to identify each of the phases examined in the main text. At each point in (Δ, R_b) space, $c_\alpha^{(2)}$ is averaged across all available snapshots (mimicking “measuring the correlation functions” from the snapshots) and then fed into the final logistic layer. Only the checkerboard (red) and star (green) phases are well resolved by this minimal model.

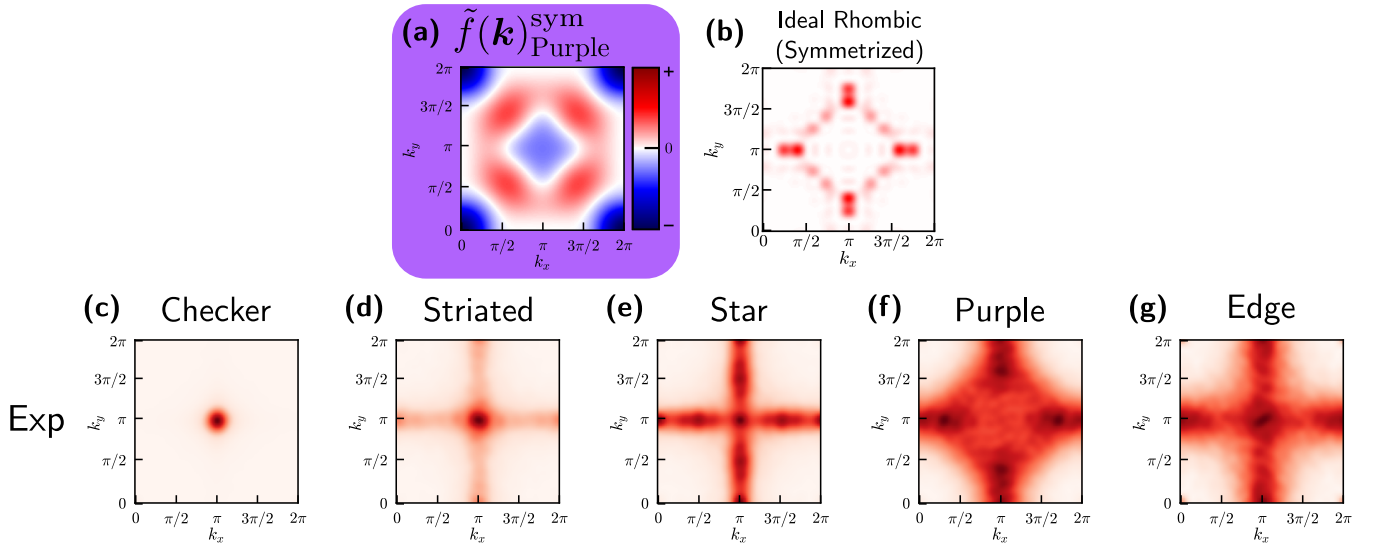


FIG. 11. (a) Fourier intensity weighting map derived from a second-order model trained to identify the purple phase. (b) Symmetrized Fourier intensities resulting from the density-normalized ideal long-range rhombic ordering. [(c)–(g)] Fourier intensities of per-site density-normalized experimental data $\delta\hat{n}(\mathbf{k})$ sampled deep in each of the identified phases.

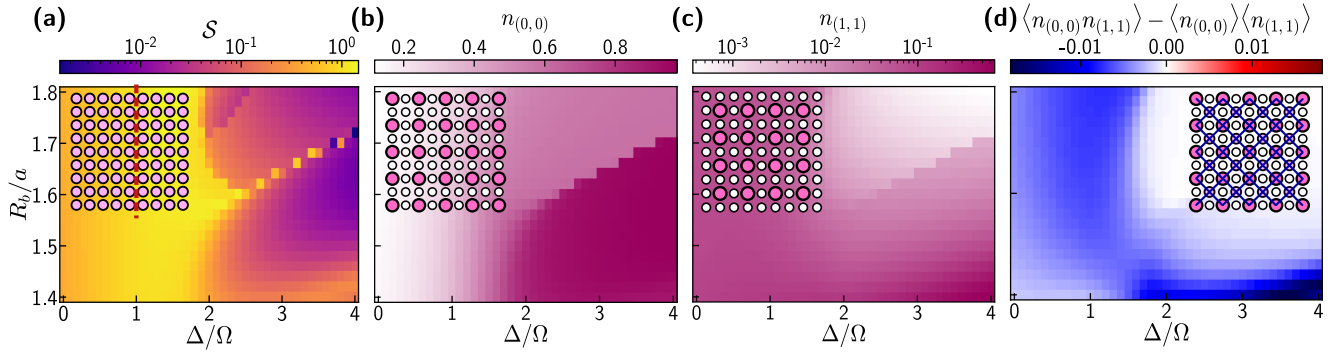


FIG. 12. Density-matrix renormalization group results for a 9×9 system. (a) Bipartite entanglement entropy $\mathcal{S}$ between two halves of the system. [(b),(c)] Average density on (0, 0) and (1, 1) sublattices, respectively. (d) Average connected density-density correlator between next-nearest-neighboring sites on the so-called (0, 0) and (1,1) sublattices, marked in pink and purple in the inset, respectively.

learns several key longer-range patterns, which help us characterize the phase.

APPENDIX D: ENTANGLEMENT GENERATED BY RYDBERG INTERACTIONS

As often exploited by Rydberg-atom quantum simulators, the Rydberg blockade effect can generate entanglement between interacting atoms [45–47]. Here, we perform simple calculations to examine the interplay between entanglement and correlations in the vicinity of the transition to the striated phase and draw connections to the correlations in the experimental data uncovered by our CCNN analysis. To this end, we perform density-matrix renormalization group (DMRG) calculations using a “snake-like” matrix product state ansatz for a 9×9 system with open boundaries. The Rydberg Hamiltonian realized by the experiment examined in this paper is given by Eq. (1), and for notational simplicity, we will refer to the first, second, and third terms therein as $\hat{\Omega}$, $\hat{\Delta}$, and $\hat{V}$, respectively. Entanglement can be generated in the ground state of $\hat{H} (\equiv H/\hbar)$ due to energetic competition between these terms. For any two sites, the interaction term $\hat{V}$ prefers small overlap with basis states containing $|rr\rangle$, while $\hat{\Delta}$ desires large overlap with all of $|gr\rangle$, $|rg\rangle$, and $|rr\rangle$. Crucially, $\hat{\Omega}$ favors weight to be present with opposite phase between basis states with a single site flipped as $|g\rangle \leftrightarrow |r\rangle$. As a result, for a two-site system, the ground state as $R_b/x_{12} \rightarrow \infty$ places weight across all of the $|gg\rangle$, $|rg\rangle$, $|gr\rangle$ basis states, but not $|rr\rangle$, resulting in an entangled, anticorrelated state.

To examine this behavior more closely, we now turn to the results of the DMRG computations. In Fig. 12, we show the bipartite entanglement entropy between two halves of the system. Within the disordered phase ($\Delta \lesssim 1$), the entanglement entropy of the ground state increases monotonically as one approaches the quantum critical points, and, at large R_b , the density on any site is anticorrelated with that of its next-nearest neighbors in the corners [Fig. 12(d)]. As we transition deep into the classically ordered phases, both the entanglement and the connected correlations vanish due to the density on each site approaching either 0 or 1 [Figs. 12(b) and 12(c)]. However, in a narrow region at $R_b/a \approx 1.4$ where quantum fluctuations stabilize a significant density on the (1,1)

sublattice [see Fig. 12(d)], both entanglement and diagonal anticorrelations survive. We emphasize that this entanglement is dependent on the state on each sublattice remaining in a quantum superposition of $|g\rangle$ and $|r\rangle$, as is uniquely characteristic of the striated phase.

Indeed, as shown in Fig. 13, in the experiment, the connected part of many short-range correlations remains finite and negative upon transitioning into the striated phase. In particular, all nearest-neighbor correlators remain anticorrelated, along with next-nearest-neighbor correlators between the two excited sublattices. However, there are many confounding effects, which make it difficult to pinpoint with certainty the origin of these correlations. First, due to decoherence and experimental noise, the experiment, in principle, produces a mixed rather than a pure state. Given a mixed state ρ , nonzero connected correlations can emerge either by entanglement, or by “classical” correlations between the pure states comprising ρ [48]. Efficient means for unambiguously revealing entanglement in experimental settings without full tomography is an active field of research [49–55].

Moreover, quasiadiabatic sweeps across phase boundaries produce final states, which are not perfect ground states but are dependent on the original state starting from which the phase boundary was crossed. As the system is theoretically transitioning from a region of high entanglement, where nearby neighbors have anticorrelated densities, it is likely that some of the magnitude of correlations captured by the CCNN is not directly representative of the true ground state, instead having been “frozen in” from before the transition [33]. Nevertheless, the nature (and, in particular, the signs) of these correlations still reveals qualitative structures of each identified phase.

APPENDIX E: SIGN STRUCTURES OF THIRD-ORDER CORRELATORS

By inspecting the learned $\beta_\alpha^{(3)}$ and the patterns in the learned associated filters f_α , we can determine which three-point correlations are being measured for a target phase, and whether they should be positive or negative within the phase. However, the sign of a three-point contribution can be somewhat confusing to interpret, as there are multiple ways to obtain positive/negative three-point correlations. The CCNN

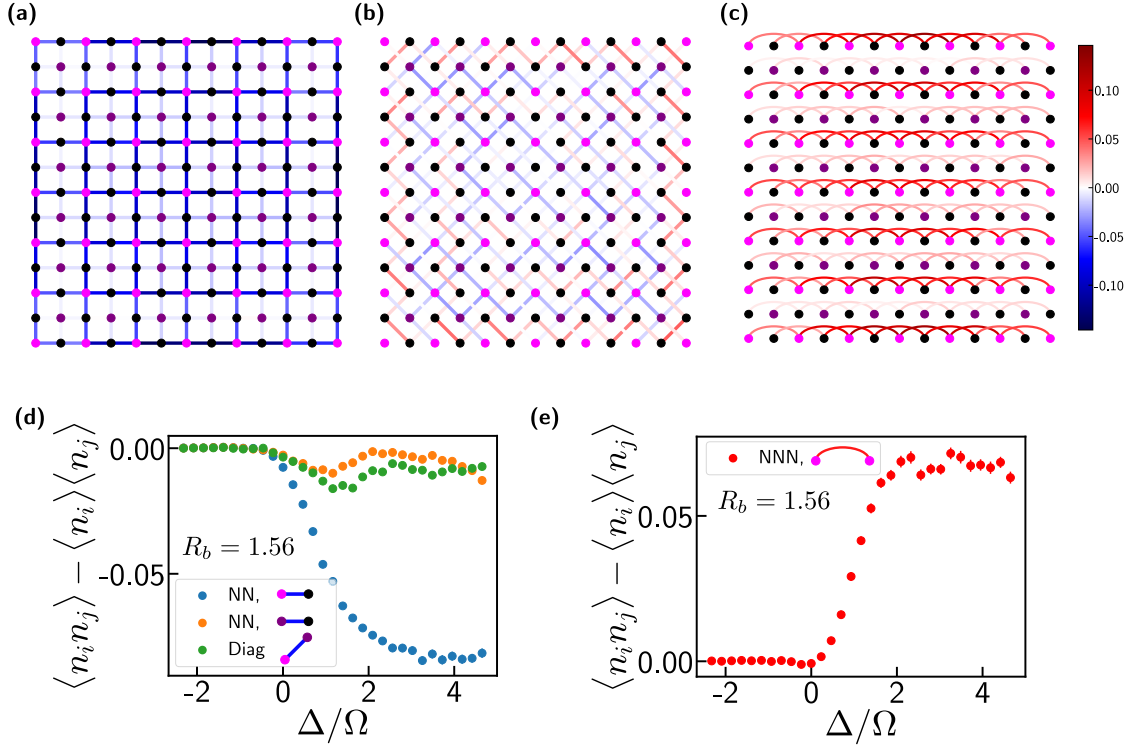


FIG. 13. [(a)–(c)] Measurements of connected correlations from the striated experimental training dataset (see Table I), performed on each individual nearest-neighbor (NN), next-nearest-neighbor (NNN), or next-to-next-nearest-neighbor (NNNN) bond, respectively. The colors of each site are for visual aid, showing the sites expected to be in the mostly excited (pink), mostly ground (purple), and entirely ground (black) states in the ideal striated limit. [(d),(e)] Averaged correlations across all bonds of different symmetry classes, tracked as a function of Δ/Ω for a cut at $R_b/a = 1.56$.

itself does not inherently point out *how* to interpret these correlators—simply that they are strongly positive/negative within the phase. Manual follow-up and investigation is always necessary to understand the underlying physics. This section attempts to clarify the subtleties of these measured

three-point correlators, and presents explicit measurements thereof from the data to confirm that the CCNN’s identification was meaningful.

Within this section, for notational brevity, spatial indices are written as (vector) subscripts. Given density-normalized

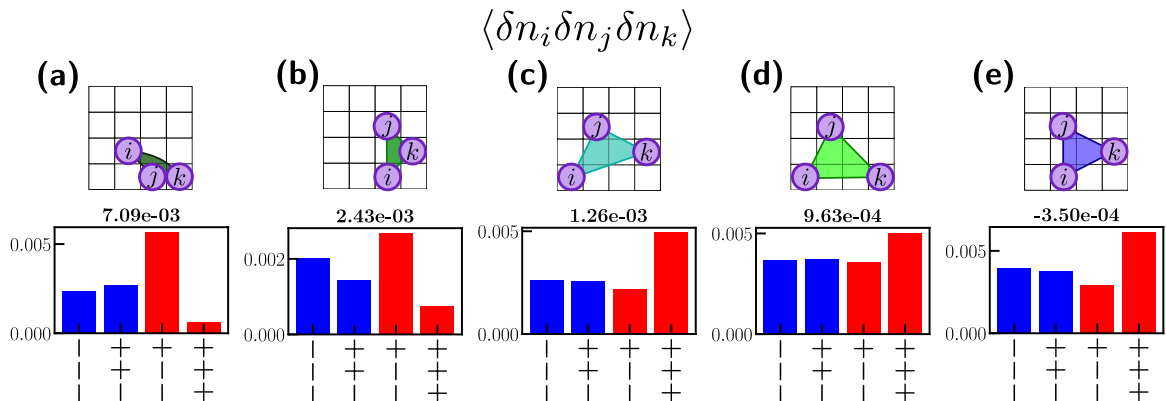


FIG. 14. [(a)–(e)] The three-point connected correlators identified from the third-order CCNN [(a)–(d)] in the main text, along with another notable group of correlations characterizing the star and rhombic phases (e). For each, we show the contributions to the correlator from three-site density fluctuation patterns of different sign configurations, measured from the snapshots in the purple phase training dataset (see Table I). Contributions are summed across all eight reflection/rotation-transformed correlators, and averaged across all translations, which leave all three sites within the snapshot. Red bars show magnitudes of positive contributions, while blue bars correspond to negative contributions. For simplicity, the bar graph shows the $(++-)$ and $(+--)$ contributions averaged over all three possible sign combinations. Above each bar graph, we show the total value of the correlator, obtained by summing all contributions with the appropriate sign and multiplying the averaged mixed-sign contributions by 3.

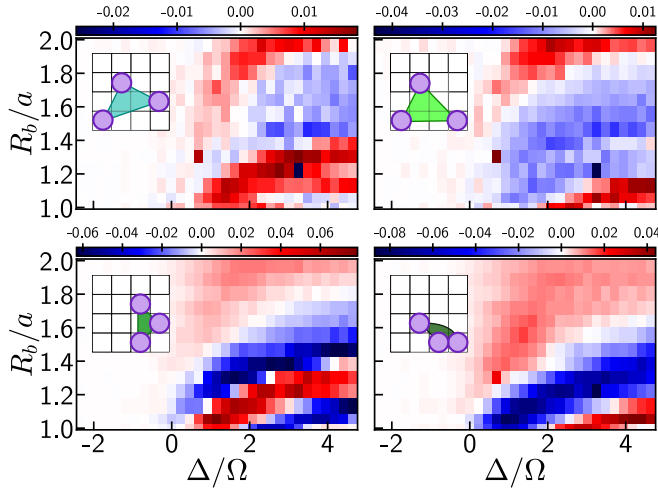


FIG. 15. Connected three-point correlators that were discovered by our CCNNs to capture the ordered phase at large R_b/a , used in tandem with other two- and three-point correlators not shown. All correlators are summed across all 8 reflection/rotation-transformed versions and averaged over all translations, which leave all three sites within the snapshot. At low R_b/a , these correlators become positive again due to different sign contributions [(+--) for the top two and (+++) for the bottom two].

snapshots $\delta n_i \equiv n_i - \langle n_i \rangle$, uniform third-order CCNN features measure weighted sums of connected three-point correlation functions, averaged over all spatial translations,

$$\langle c_\alpha^{(3)} \rangle = \sum_x \sum_{ijk=(0,0)}^{(L_f-1, L_f-1)} f_i f_j f_k \langle \delta n_{x+i} \delta n_{x+j} \delta n_{x+k} \rangle, \quad (\text{E1})$$

where the inner sum runs over all configurations of displacements (i, j, k) within the spatial extent of the filter f of length L_f . Due to the symmetrization process outlined in Appendix B 1, this must also be averaged over all rotations and flips of the three-point pattern.

In terms of individual density configurations, each of these correlators acquire positive contributions either when all of $(\delta n_i, \delta n_j, \delta n_k)$ are positive (+++), or when only one is

(+--), (-+-), (--+). The CCNN does not directly inform us as to how the correlator became positive—manual follow-up is necessary to uncover this information. For example, in Fig. 14, we show the statistics of these different sign contributions for the dominant three-point correlators learned by the CCNN within the purple region. For conciseness, the heights of the bars corresponding to mixed-sign contributions are averaged over all configurations, which produce the same sign. In Fig. 15, we show the value of a handful of these key correlators across the $(\Delta/\Omega, R_b/a)$ parameter space in the data.

From this, we can see that the short-range patterns learned by the CCNN are actually producing positive signals due to a large number of (+--) density configurations, as shown in Figs. 14(a) and 14(b). This points to the longer-range packing of Rydberg excitations within this phase—if one of the sites in the indicated triples is excited, it is more likely that the other two are in the ground state. Meanwhile, the patterns of Figs. 14(c) and 14(d) are positive dominantly due to (+++) configurations, indicating that these motifs signal actual common configurations of joint Rydberg excitations. Figure 14 shows that the star-like configurations also have a large (+++) signal in this phase, as expected from the idealized pattern, but the other sign contributions cause this correlator to turn negative and not be picked up by the CCNN. Together, we can infer from these correlations that we are probing a (possibly rhombic-like) ordered phase, which is failing to develop long-range order due to the incommensurate geometry of the system. Similar arguments apply to the motifs for the staggered and nematic phases as well.

In Fig. 15, we show the extent in parameter space of the key identified connected three-point correlators, and observe that the purple region colored by the CCNN does indeed correspond to the region where the long-range three-point motifs uniquely produce a positive signal. Further theoretical and numerical analysis is needed to confirm that this phase exists in the thermodynamic limit and that the rough region of parameter space identified by our CCNN corresponds to the true region hosting this phase, as well as to determine if these higher-order signals remain good indicators of the phase in the thermodynamic limit.

-
- [1] P. Schauß, M. Cheneau, M. Endres, T. Fukuhara, S. Hild, A. Omran, T. Pohl, C. Gross, S. Kuhr, and I. Bloch, Observation of spatially ordered structures in a two-dimensional Rydberg gas, *Nature (London)* **491**, 87 (2012).
 - [2] A. Browaeys and T. Lahaye, Many-body physics with individually controlled Rydberg atoms, *Nat. Phys.* **16**, 132 (2020).
 - [3] H. Labuhn, D. Barredo, S. Ravets, S. de Léséleuc, T. Macrì, T. Lahaye, and A. Browaeys, Tunable two-dimensional arrays of single Rydberg atoms for realizing quantum Ising models, *Nature (London)* **534**, 667 (2016).
 - [4] H. Bernien, S. Schwartz, A. Keesling, H. Levine, A. Omran, H. Pichler, S. Choi, A. S. Zibrov, M. Endres, M. Greiner *et al.*, Probing many-body dynamics on a 51-atom quantum simulator, *Nature (London)* **551**, 579 (2017).
 - [5] S. Ebadi, T. T. Wang, H. Levine, A. Keesling, G. Semeghini, A. Omran, D. Bluvstein, R. Samajdar, H. Pichler, W. W. Ho *et al.*, Quantum phases of matter on a 256-atom programmable quantum simulator, *Nature (London)* **595**, 227 (2021).
 - [6] K. Vogel and H. Risken, Determination of quasiprobability distributions in terms of probability distributions for the rotated quadrature phase, *Phys. Rev. A* **40**, 2847 (1989).
 - [7] D. F. V. James, P. G. Kwiat, W. J. Munro, and A. G. White, Measurement of qubits, *Phys. Rev. A* **64**, 052312 (2001).
 - [8] D. Gross, Y.-K. Liu, S. T. Flammia, S. Becker, and J. Eisert, Quantum State Tomography via Compressed Sensing, *Phys. Rev. Lett.* **105**, 150401 (2010).
 - [9] R. Samajdar, W. W. Ho, H. Pichler, M. D. Lukin, and S. Sachdev, Complex Density Wave Orders and Quantum Phase

- Transitions in a Model of Square-Lattice Rydberg Atom Arrays, *Phys. Rev. Lett.* **124**, 103601 (2020).
- [10] M. D. Lukin, M. Fleischhauer, R. Cote, L. M. Duan, D. Jaksch, J. I. Cirac, and P. Zoller, Dipole Blockade and Quantum Information Processing in Mesoscopic Atomic Ensembles, *Phys. Rev. Lett.* **87**, 037901 (2001).
- [11] C. Miles, A. Bohrdt, R. Wu, C. Chiu, M. Xu, G. Ji, M. Greiner, K. Q. Weinberger, E. Demler, and E.-A. Kim, Correlator convolutional neural networks as an interpretable architecture for image-like quantum matter data, *Nat. Commun.* **12**, 3905 (2021).
- [12] J. Carrasquilla and R. G. Melko, Machine learning phases of matter, *Nat. Phys.* **13**, 431 (2017).
- [13] S. J. Wetzel, Unsupervised learning of phase transitions: From principal component analysis to variational autoencoders, *Phys. Rev. E* **96**, 022140 (2017).
- [14] W. Hu, R. R. P. Singh, and R. T. Scalettar, Discovering phases, phase transitions, and crossovers through unsupervised machine learning: A critical examination, *Phys. Rev. E* **95**, 062122 (2017).
- [15] E. Greplova, A. Valenti, G. Boschung, F. Schäfer, N. Lörch, and S. D. Huber, Unsupervised identification of topological phase transitions using predictive models, *New J. Phys.* **22**, 045003 (2020).
- [16] K. Liu, J. Greitemann, and L. Pollet, Learning multiple order parameters with interpretable machines, *Phys. Rev. B* **99**, 104410 (2019).
- [17] Y.-H. Liu and E. P. L. van Nieuwenburg, Discriminative Cooperative Networks for Detecting Phase Transitions, *Phys. Rev. Lett.* **120**, 176401 (2018).
- [18] J. Venderley, K. Mallayya, M. Matty, M. Krogstad, J. Ruff, G. Pleiss, V. Kishore, D. Mandrus, D. Phelan, L. Poudel, A. G. Wilson, K. Weinberger, P. Upreti, M. Norman, S. Rosenkranz, R. Osborn, and E.-A. Kim, Harnessing interpretable and unsupervised machine learning to address big data from modern x-ray diffraction, *Proc. Natl. Acad. Sci.*, **119**, e2109665119 (2022).
- [19] N. Käming, A. Dawid, K. Kottmann, M. Lewenstein, K. Sengstock, A. Dauphin, and C. Weitenberg, Unsupervised machine learning of topological phase transitions from experimental data, *Mach. Learn.: Sci. Technol.* **2**, 035037 (2021).
- [20] J. Arnold, F. Schäfer, M. Žonda, and A. U. J. Lode, Interpretable and unsupervised phase classification, *Phys. Rev. Res.* **3**, 033052 (2021).
- [21] K. Kottmann, P. Huembeli, M. Lewenstein, and A. Acín, Unsupervised Phase Discovery with Deep Anomaly Detection, *Phys. Rev. Lett.* **125**, 170603 (2020).
- [22] A. Cole, G. J. Loges, and G. Shiu, Quantitative and interpretable order parameters for phase transitions from persistent homology, *Phys. Rev. B* **104**, 104426 (2021).
- [23] P. Huembeli, A. Dauphin, P. Wittek, and C. Gogolin, Automated discovery of characteristic features of phase transitions in many-body localization, *Phys. Rev. B* **99**, 104106 (2019).
- [24] C. Casert, T. Vieijra, J. Nys, and J. Ryckebusch, Interpretable machine learning for inferring the phase boundaries in a nonequilibrium system, *Phys. Rev. E* **99**, 023304 (2019).
- [25] P. Huembeli, A. Dauphin, and P. Wittek, Identifying quantum phase transitions with adversarial neural networks, *Phys. Rev. B* **97**, 134109 (2018).
- [26] B. S. Rem, N. Käming, M. Tarnowski, L. Asteria, N. Fläschner, C. Becker, K. Sengstock, and C. Weitenberg, Identifying quantum phase transitions using artificial neural networks on experimental data, *Nat. Phys.* **15**, 917 (2019).
- [27] C. M. Bishop, *Pattern Recognition and Machine Learning*, Information Science and Statistics (Springer, New York, 2006).
- [28] J. Venderley, V. Khemani, and E.-A. Kim, Machine Learning Out-of-Equilibrium Phases of Matter, *Phys. Rev. Lett.* **120**, 257204 (2018).
- [29] A. Bohrdt, C. S. Chiu, G. Ji, M. Xu, D. Greif, M. Greiner, E. Demler, F. Grusdt, and M. Knap, Classifying snapshots of the doped Hubbard model with machine learning, *Nat. Phys.* **15**, 921 (2019).
- [30] R. Meyes, M. Lu, C. W. de Puiseau, and T. Meisen, Ablation studies in artificial neural networks, [arXiv:1901.08644](https://arxiv.org/abs/1901.08644).
- [31] M. J. O'Rourke and G. K.-L. Chan, Entanglement in the quantum phases of an unfrustrated Rydberg atom array, [arXiv:2201.03189](https://arxiv.org/abs/2201.03189).
- [32] M. A. Metlitski, C. A. Fuertes, and S. Sachdev, Entanglement entropy in the $O(N)$ model, *Phys. Rev. B* **80**, 115122 (2009).
- [33] W. H. Zurek, U. Dorner, and P. Zoller, Dynamics of a Quantum Phase Transition, *Phys. Rev. Lett.* **95**, 105701 (2005).
- [34] M. Metlitski, Boundary criticality of the $O(N)$ model in $d = 3$ critically revisited, *SciPost Phys.* **12**, 131 (2022).
- [35] M. Kalinowski, R. Samajdar, R. G. Melko, M. D. Lukin, S. Sachdev, and S. Choi, Bulk and boundary quantum phase transitions in a square Rydberg atom array, *Phys. Rev. B* **105**, 174417 (2022).
- [36] G. Semeghini, H. Levine, A. Keesling, S. Ebadi, T. T. Wang, D. Bluvstein, R. Verresen, H. Pichler, M. Kalinowski, R. Samajdar *et al.*, Probing topological spin liquids on a programmable quantum simulator, *Science* **374**, 1242 (2021).
- [37] R. Samajdar, W. W. Ho, H. Pichler, M. D. Lukin, and S. Sachdev, Quantum phases of Rydberg atoms on a kagome lattice, *Proc. Natl. Acad. Sci. USA* **118**, e2015785118 (2021).
- [38] R. Verresen, M. D. Lukin, and A. Vishwanath, Prediction of Toric Code Topological Order from Rydberg Blockade, *Phys. Rev. X* **11**, 031005 (2021).
- [39] R. Samajdar, D. G. Joshi, Y. Teng, and S. Sachdev, Emergent $\mathbb{Z}_2$ gauge theories and topological excitations in Rydberg atom arrays, [arXiv:2204.00632](https://arxiv.org/abs/2204.00632).
- [40] M. Fishman, S. R. White, and E. M. Stoudenmire, The ITensor software library for tensor network calculations, [arXiv:2007.14822](https://arxiv.org/abs/2007.14822).
- [41] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, PyTorch: An imperative style, high-performance deep learning library, in *Advances in Neural Information Processing Systems* 32, edited by H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Curran Associates, Inc., New York, 2019), pp. 8024–8035.
- [42] github.com/KimGroup/QGasML.
- [43] D. P. Kingma and J. Ba, Adam: A method for stochastic optimization, [arXiv:1412.6980](https://arxiv.org/abs/1412.6980).
- [44] S. Ioffe and C. Szegedy, Batch normalization: Accelerating deep network training by reducing internal covariate shift, [arXiv:1502.03167](https://arxiv.org/abs/1502.03167).

- [45] M. Saffman, T. G. Walker, and K. Mølmer, Quantum information with Rydberg atoms, *Rev. Mod. Phys.* **82**, 2313 (2010).
- [46] H. Weimer, M. Müller, I. Lesanovsky, P. Zoller, and H. P. Büchler, A Rydberg quantum simulator, *Nat. Phys.* **6**, 382 (2010).
- [47] T. Wilk, A. Gaëtan, C. Evellin, J. Wolters, Y. Miroshnychenko, P. Grangier, and A. Browaeys, Entanglement of Two Individual Neutral Atoms Using Rydberg Blockade, *Phys. Rev. Lett.* **104**, 010502 (2010).
- [48] V. Vedral, M. B. Plenio, M. A. Rippin, and P. L. Knight, Quantifying Entanglement, *Phys. Rev. Lett.* **78**, 2275 (1997).
- [49] A. Elben, R. Kueng, H.-Y. R. Huang, R. van Bijnen, C. Kokail, M. Dalmonte, P. Calabrese, B. Kraus, J. Preskill, P. Zoller, and B. Vermersch, Mixed-State Entanglement from Local Randomized Measurements, *Phys. Rev. Lett.* **125**, 200501 (2020).
- [50] H.-Y. Huang, R. Kueng, and J. Preskill, Predicting many properties of a quantum system from very few measurements, *Nat. Phys.* **16**, 1050 (2020).
- [51] A. Ketterer, N. Wyderka, and O. Gühne, Entanglement characterization using quantum designs, *Quantum* **4**, 325 (2020).
- [52] L. Knips, J. Dziewior, W. Kłobus, W. Laskowski, T. Paterek, P. J. Shadbolt, H. Weinfurter, and J. D. A. Meinecke, Multipartite entanglement analysis from random correlations, *npj Quantum Inf.* **6**, 51 (2020).
- [53] A. Neven, J. Carrasco, V. Vitale, C. Kokail, A. Elben, M. Dalmonte, P. Calabrese, P. Zoller, B. Vermersch, R. Kueng, and B. Kraus, Symmetry-resolved entanglement detection using partial transpose moments, *npj Quantum Inf.* **7**, 152 (2021).
- [54] Y. Zhou, P. Zeng, and Z. Liu, Single-Copies Estimation of Entanglement Negativity, *Phys. Rev. Lett.* **125**, 200502 (2020).
- [55] X.-D. Yu, S. Imai, and O. Gühne, Optimal Entanglement Certification from Moments of the Partial Transpose, *Phys. Rev. Lett.* **127**, 060504 (2021).